AD-A095 537 NAVAL RESEARCH LAB WASHINGTON DC F/G 22/2
LORI: A MULTISATELLITE, MULTISYSTEM, TIME-COVERAGE ANALYSIS PRO--ETC(U)
FEB 81 W H HARR
UNCLASSIFIED NRL-8448 NL

1 OF 2
AD A
095537

A

55

(12)

(14) NRL-[redacted] 8448

(6)

LORI: A Multisatellite, Multisystem, Time-Coverage Analysis Program

LEVEL II

(10) WILLIAM H. HARR

Systems Research Branch
Space Systems Division

(9) Final rept.,

(11) 25 Feb[redacted]81

(12) 114

(16) RRϕ14ϕ2

(17) RRϕ14ϕ241

61153N

NAVAL RESEARCH LABORATORY
Washington, D.C.

251950

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 8448 ✓	2. GOVT ACCESSION NO. AD-A095 537	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) LORI: A MULTISATELLITE, MULTISYSTEM, TIME-COVERAGE ANALYSIS PROGRAM		5. TYPE OF REPORT & PERIOD COVERED Final report on one phase of a continuing problem. 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) William H. Harr		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61153N; RR0140241; 79-0722-0-0
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE February 25, 1981 13. NUMBER OF PAGES 113
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Satellite systems	Timeliness
Ground coverage	Revisit times
Simultaneous coverage	Antenna patterns
Coverage analysis	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

It is important to be able to describe the ground-coverage performance of multiple-satellite systems, whether existing or proposed. Measures of delays between sightings of ground targets, and simultaneous sightings, are useful, whether for a single system or between several separate multi-satellite systems. The ground coverage may involve areas, points, grids, or moving targets. The program LORI is designed to handle these problems.

(Continues)

20. Abstract (Continued)

LORI (Locater of Repeating Incidents) is a group of programs designed for analysis of the effectiveness of satellite systems in coverage of areas of the earth. Its principal features include speed of operation, ease of modification, large capabilities, and simple installation on small or large computer systems. LORI will generate statistics (or specific results) on revisit times and simultaneous coverage for one or two systems of satellites, each system consisting of any number of satellites. These statistics can then be plotted as functions of time, latitude, longitude, or other variables.

Accession For		
NTIS GRA&I	X	
DTIC TAB	☐	
Unannounced	☐	
Justification		
By		
Distribution/		
Availability Codes		
Dist	Avail and/or Special	
A		

CONTENTS

CONTENTS (Continued)

LORI: A MULTISATELLITE, MULTISYSTEM, TIME-COVERAGE ANALYSIS PROGRAM

1.0 PROGRAM DESCRIPTION

1.1 Overview

LORI (Locater of Repeating Incidents) is a general-purpose time-coverage program. In its basic form it will generate simultaneous coverage and/or revisit delay times for one or two systems of satellites to selected ground targets. With minor changes the program can handle any number of systems. Associated programs can then use the output from LORI to generate and plot statistics that describe the effectiveness of the system or systems. The power of LORI results from its speed, modular structure, ease of modification for specific systems, capability of handling very large systems of satellites, and its ability to do all of these on very small to very large computer systems with minimal change.

Since part of LORI's usefulness results from its ability to adapt to the needs of the user, the description of the inner workings of the program will be more detailed than usual. Due to LORI's modularity, the user usually need concern himself only with changing one section of LORI at a time for some specific modification. The program listing contains extensive comments describing input, output, and the algorithms used. The user is cautioned, however, that most of the following sections do not need to be read or understood in order to run LORI as it stands. Specifically, only sections 1, 2, and 3 should be needed to know how to run LORI and understand the input and output. Sections 4 through 8 are to help the user make modifications to LORI.

LORI reads a single input file that contains the satellite elements, the period of time the run is to cover, the time step of the run, keywords that describe the type of ground targets the satellites are to look at, and the format of the output. LORI then propagates all of the satellites through their orbits for the time period using an analytic orbit generator that includes the secular effects of the earth's oblateness (J_2). At each time step, all of the satellites are checked to determine what ground targets they can see. The conditions for a sighting include the satellite antenna parameters such as elevation or cone angle and auxiliary conditions such as the simultaneous sighting of a ground station. At the end of this part of the run, LORI has generated an internal table referenced by target and satellite that contains the start and stop times of all of the sightings of each target by each satellite. Associated tables can contain information such as the ground station visible for each sighting. At this point the postprocessor sections can process this table to determine revisit times, simultaneous inter- and intrasystem sightings, and so on. Depending on keyword input, LORI will then write out to a single file, or one file to a target, the information requested for each target. Output is to files FOR008.DAT, FOR009.DAT, FOR010.DAT, etc., one per target, unless all output is to a single file, FOR008.DAT. If the targets are a grid, the program will process as many targets as it is dimensioned for at the same time, and write the results out to the above file or files. It will then start over with a new set of targets in the grid and write out to FOR008.DAT;2, FOR009.DAT;2, etc., and so on, until all targets in the grid are covered.

The calling structure of LORI is shown in Fig. 1. This figure can also be used as the overlay structure for small computer systems.

Manuscript submitted August 10, 1980.

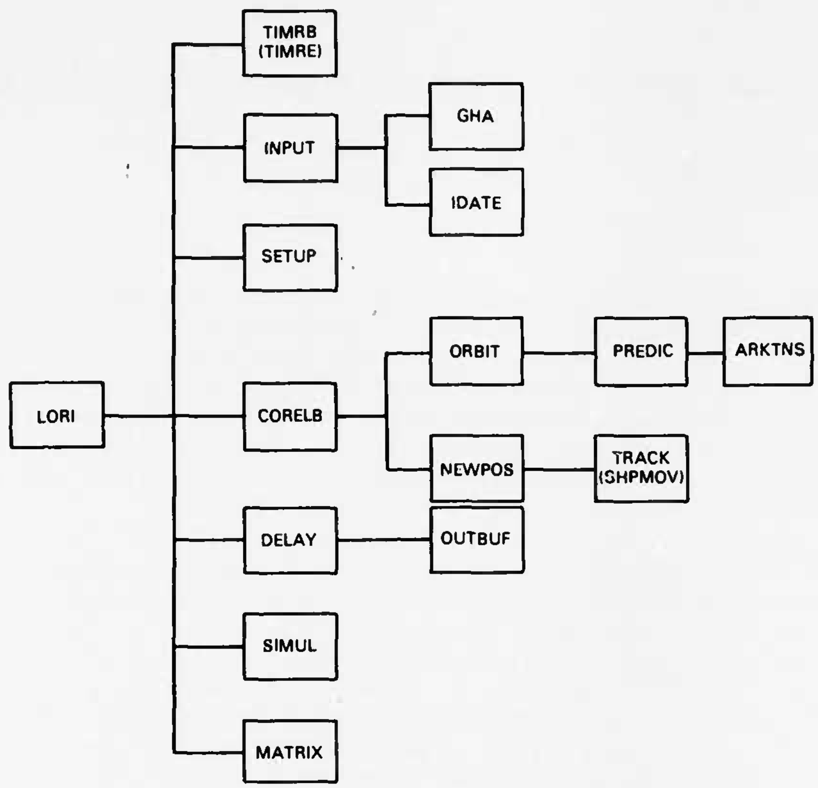

Fig. 1 — Calling structure for the LORI program

For specified needs, such as simultaneous coverage for more than two satellite systems, additional postprocessors can be written and added with little or no change to the rest of the program.

1.2 General Capabilities

In the same run, the program can handle specified input targets, a grid of targets, targets moving along a great circle, and targets whose movements are specified in a file. The targets may be points or areas, where the satellites may be required to see all or any of the areas; there can be one or two systems of satellites; the satellites may have completely independent elements from other satellites in each system; there can be an arbitrary number of satellites in each system; each satellite can have its own antenna parameters; there can be up to six separate types of antennas; or satellites can require ground stations, with separate antenna parameters for the ground stations.

The output can be in the following forms: number of sightings by each satellite for each target; duration of each sighting by each satellite for each target; delays between sightings of each target by two systems of satellites; simultaneous viewings of targets by two systems of satellites; total time the target is seen by each system of satellites; graphic (printer) output of the sighting of each target by all satellites. Also, the output may be on separate files for each target or all on one file. For most forms of output the user can specify the explicit output of individual cases or a short form with totals.

1.3 Antenna Types

LORI can currently handle six different antenna types. The types are input separately for each satellite and are specified by number as: 1 = elevation angle (ϵ), 2 = co-dip angle (α), 3 = earth central half-angle (θ), 4 = half swath width in nautical miles (D), 5 = annular ring (α_1, α_2), 6 = side-looking beam (α,β). Types 1 through 4 use one input parameter to describe the antenna footprint, and types 5 and 6 use two input parameters. Figure 2 describes the parameters used for types 1-4, Fig. 3 describes the parameters for type 5, and Fig. 4 describes the parameters for type 6. The ground swath described by the earth central half-angle and the half swath width, types 3 and 4, does not change in size as the satellite increases or decreases its altitude (due to eccentricity). The swath will always be the same. For the other antenna types the ground swath will change in size. For elevation angle input, it will vary as

$$\theta = \cos^{-1} (\cos (\epsilon) R_e/R) - \epsilon$$

where

θ is the earth central half-angle
ϵ is the elevation angle
R is the magnitude of the position vector
R_e is the earth radius.

For co-dip angle input the coverage will vary as

$$\theta = \sin^{-1}(\sin (\alpha) R/R_e) - \alpha$$

where α is the co-dip angle (90° - dip angle). The annular ring uses the co-dip angle definition. The parameters input are the outer and inner co-dip angles (α_1, α_2).

The side-looking antenna beam uses as input the co-dip angle of the boresight of the beam, and the beam width (α,β). If the boresight is positive, the beam's azimuth is 270° (port); if it is negative, the azimuth is 90° (starboard). The beam has only one dimension, and it defines two points, the far

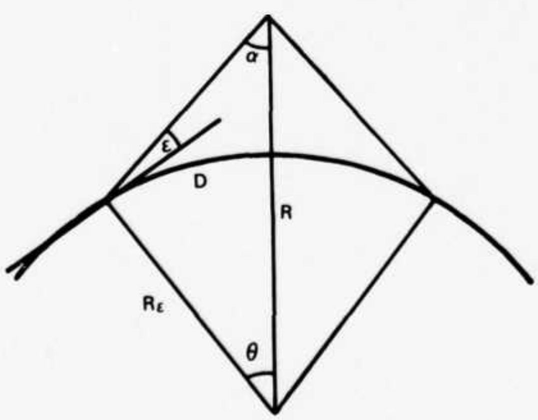

Fig. 2 — Antenna parameters

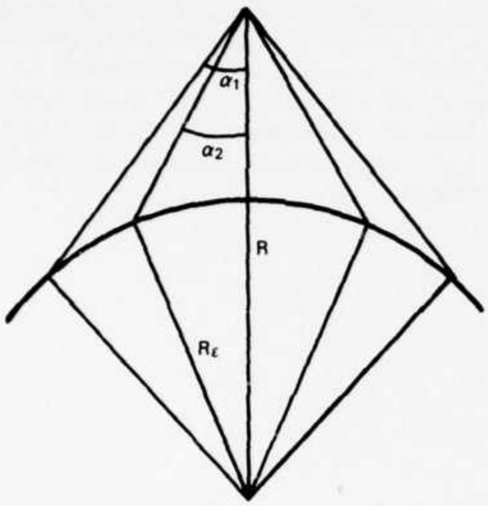

Fig. 3 — Antenna parameters for annular antennas

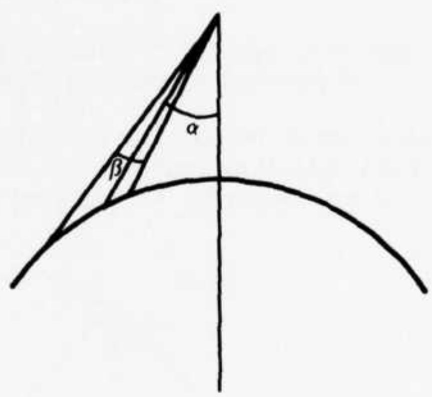

Fig. 4 — Antenna parameters for sidelooking antenna

and near points of intersection of the beam with the earth's surface. When the satellite is close enough to a target for a sighting to occur, the program calculates these two points for each time step. The current two points and the last two points define an approximate "rectangle" on the earth's surface. The program then decides whether the target is within this rectangle. Note that this results in a sighting duration of only one time step. In effect, for this type of antenna the beam is one-dimensional. This method will not be able to handle cases where the swath covers the poles.

1.4 Ground Stations

For satellites that require having a ground station in view at the same time as a target, the parameter that determines whether the satellite can see the ground station is the elevation angle of the satellite as seen from the ground station. The minimum elevation angle is a parameter associated with each ground station, and is input with each ground station's coordinates.

1.5 Target and Ground Station Radius

The program can currently handle one additional variable associated with a satellite seeing a target or ground station. This variable is the radius of the target or ground station. The radius is input with the target or ground station's other elements, and is the earth central half-angle, in degrees. The radius defines a circle centered on the target or ground station. If the radius is positive, the satellite must see all of the circle to register a sighting; if negative, the satellite can see any of the circle and register a sighting. A radius of zero defines the target or ground station as a point.

1.6 Postprocessors

Once LORI has created its internal table of sightings of each target by each satellite, the user can specify several different postprocessors to be run that will process the table according to different criteria. The output from these processors provides an aid to analysis and evaluation of the satellite systems. A description of the current postprocessors follows.

1.6.1 DELAY

The DELAY postprocessor is designed for calculating the delays between sightings by two systems of satellites. The satellites in system 1 are so specified by being the first satellites input (from 1 to NSAT1). The delays calculated are from a system 1 satellite sighting to a later system 2 satellite sighting. A minimum and maximum allowable delay can be specified. Normally a delay will be calculated as from the start of the system 1 sighting to the start of the system 2 sighting. If that definition causes the delay to be shorter than the minimum delay, the program will check whether the system 2 sighting duration was long enough such that a later time, still during the sighting, will cause a delay long enough to be greater than the minimum. Similarly, if the delay calculated is longer than the maximum, the program will check if a later system 1 time will shorten the delay to the point that the delay is shorter than the maximum allowable. The processor will allow multiple revisits by system 2 to the same system 1 sighting. Output from the processor can be a short form, with the statistics of the run, or the complete output in the form, for each delay, of times, durations, satellites involved, and ground stations used.

1.6.2 SIMUL

The SIMUL postprocessor calculates the times both systems of satellites can see the targets at the same time. Specifically, it will calculate when any satellite of system 1 and any satellite of system 2 both see a target at the same time. The program can print out the data for each simultaneous event, or a short form with just the statistics of the run. The processor can also print out the total times each satellite saw each target, the total times each system saw each target, the individual times each system saw each target, and the individual times both systems saw each target. SIMUL can be run with one or two systems. When run with one system, there will be no simultaneous printout, but the processor will print out the merged coverage time for the one system. This is useful for determining such statistics as the total time a system sees a target, total time it does not see a target, maximum, average, and minimum delays between sightings, etc.

1.6.3 MATRIX

The MATRIX postprocessor provides a simple printer plot of the coverage of each target by each satellite vs time. The user defines a time period in terms of the step size of the program, the step number to start the period from, the number of periods the user wishes displayed, and a minimum acceptable fraction of the period. The processor will then print out, for each time period, each satellite that saw the target for at least the minimum fraction of the period that was input. The total number of satellites that saw the target for each period, and the total number of periods each satellite sees the target are also printed out.

For example, a user might want to observe the coverage one day into a run, every hour for one day, and require a satellite to see the target at least 40% of the hour to register as a sighting. With a 1-min time step, the input to MATRIX would be 1440 for the time step to start (1*24*60), 60 for the length of the period (60 time steps), 24 for the number of periods, and 0.4 for the minimum acceptable fraction. The sample input and output section gives an example of a MATRIX run, along with some caveats about its use.

This output is useful for determining which satellites are most effective for particular targets and for determining which satellites in a proposed system could be rephased to maximize coverage and minimize delays.

2.0 DESCRIPTIONS OF INPUT VARIABLES

Upper case input is literal, lower case input is to be numeric. The input routine uses free-field input, also known as list-directed input/output (I/O). This means that multiple input on the same line can be separated by spaces, commas, or carriage returns, and will be converted to the correct internal data type automatically. The input is listed here as the program expects to read it; i.e., if the input is listed on one line, the program is capable of reading it all on one line. This is not necessary, and the input for one line can be broken up into several lines if needed.

2.1 Mandatory Inputs

1. dt	Time step in minutes (default) or seconds (keyword SECONDS) [> 0.].
2. yy mm dd hh mm sss.	Start time of run [> Jan. 0, 1950, < Jan. 1, 2000].
3. yy mm dd hh mm sss.	Stop time of run [> Jan. 0, 1950, < Jan. 1, 2000].
4. nsat1	Number of satellites in system 1 [1 to MAXSAT]. Recalculated for VARIATIO.
5. nsat2	Number of satellites in system 2 [0 to (MAXSAT - NSAT1)].
6. ngs	Number of ground stations [0 to MAXG - MAXGP].
7. gslat1, gslon1, gsrad1 • • • • • • gslatn,gslonn,gsradn	Ground station elements. latitude, longitude, and radius, all in degrees. radius > 0; satellite must see all of area radius = 0; target is a point radius < 0; satellite can see any part of area, and radius is an earth central angle
8. Satellite elements as follows: a,e,i,m,ω,Ω,yy,mm,dd,hh,mm,sss., g.sta. flag,antenna code, antenna parameter one (antenna parameter two)	Elements for an individual satellite may be on more than one line (antenna parameters must start on a new line). System 1 satellites are input first. Input elements may be inertial (default) or earth-fixed. a = semimajor axis [km] e = eccentricity i = inclination [deg] m = mean anomally [deg] (= true anomally with keyword TRUE) ω = argument of perigee [deg] Ω = Right ascension of ascending node [deg] (= longitude of node with keyword LONG) Epoch follows. Epoch of satellite may be before or after epoch of program run. g. sta. flag = flag for whether this satellite needs to see a ground station at the same time it sees a target. 0 = no, 1 = yes Antenna code = index of antenna type. This determines the type of antenna the satellite has and determines the meaning of the antenna parameter.

Code = 1, parameter = elevation angle. Satellite must be above this angle as seen from the target.

Code = 2, parameter = co-dip angle. Angle is from subsatellite point to edge of antenna beam.

Code = 3, parameter = earth central half-angle.
Angle is from subsatellite point to edge of beam on the earth's surface, measured from the center of the earth (ground coverage will not vary with change of altitude of satellite).

Code = 4, parameter = half swath width (in nautical miles).

This also will not vary as the satellite's altitude does.

Code = 5, antenna is an annulus.
The two antenna parameters define the outside and inside co-dip angles of an annular cone that looks straight down.

Code = 6, antenna is a side-looking antenna. The two parameters define the co-dip angle of a boresight and the beamwidth of a narrow-beam side-looking antenna.

9. (keywords) — See below.

10. START — Must be last card in deck.

2.2 Optional Inputs (Keywords and Associated Inputs)

1. SECONDS — dt, tmin, tmax are in seconds.

2. SINGLFIL — Print all output to FOR008.DAT.; otherwise, the output for each target goes to a separate file.

3. MEAN — Satellite anomaly is mean anomaly. Default is mean anomaly.

4. TRUE — Satellite anomaly is true anomaly. Default is mean anomaly.

5. VARIATION
index start delt num — Create a grid of satellite elements that will be the same except for element # INDEX. That element will be START for satellite 1 and be START+(NUM−1)*DELT for satellite NUM.
[0 < index < 7]
[0 < num < MAXSAT+1]

6. TARGRID
blat blon elat elon spalat spalon radii

Target elements are a grid.
For this program, a grid is set of targets that are independent, but are input with one command. They will be spaced equally apart in longitude and in latitude. Note that this means that they will not be equally spaced in true distance.
The true separation of the targets in longitude will diminish as the latitude approaches ±90°.

blat = beginning latitude of grid
blon = beginning longitude of grid
elat = ending latitude of grid
elon = ending longitude of grid
spalat = spacing in latitude of grid
spalon = spacing in longitude for grid
radii = radius each point in grid will have
radii = 0; each target is a point
radii < 0; each target is an area, and satellite may see any of area
radii > 0; satellite must see all of area

7. TARINPUT — Target elements are individually input
numtarg — Number of targets input
lat1 long1 radius1
•
•
•
lat N long N radius N

8. MOVETARG — Targets are moving targets
numtarg — Number of moving targets
target #1 filename — Positions read from files whose names follow.
•
•
target #nfilename

9. SHPTRK — Targets move along great circles.
numtarg — Number of great circle targets
blat1 blon1 elat1 elon1 speed 1
•
•
•
•
•
•
•
blatn blonn elatn elonn speedn

blat1 = beginning latitude of target 1
blon1 = beginning longitude of target 1
elat1 = end latitude of target 1
elon1 = end longitude of target 1
speed1 = speed of target 1 in knots
If speed is input as 0 it is calculated by the program so the target reaches the end point at end of run.
Otherwise, if the target reaches its destination before the end of the run, the program will stop processing that target, but will continue with the other targets.

10.	DELAY	Run DELAY postprocessor.
	inc	Number of revisits of system 2 to system 1 [1 < = inc < = 20].
	tmin	Minimum revisit time [tmin > 0].
	tmax	Maximum revisit time [tmax > 0]. Tmax, tmin are in minutes unless keyword SECONDS has been used.
11.	DEBUGOUT	Print out tables GS and IGS for all satellites and targets. GS is the table of the start time of each sighting and the first time after that time that the satellite does not see the target. IGS is an associated table that holds the ground station number for that sighting (it could be used to store other data, such as maximum elevation angle). GS and IGS are referenced by satellite and target.
12.	NUMHIT	Print out the total number of hits of each target for each satellite.
13.	SIMUL	Run SIMUL postprocessor. SIMUL will compute the total time that two systems see targets simultaneously, and will print out the individual overlap sightings, if desired. It also prints the totals for the individual satellites.
14.	SHORTOUT	For *SIMUL* and *DELAY*, just print out statistics totals.
15.	SYSTMCOV	For SIMUL, print out system 1 and system 2 coverage times.
16.	TOTAL	For SIMUL, this command merges system 1 and system 2 sightings and prints out times, durations, and delays. With SHORTOUT it will just print the statistics rather than each sighting.
17.	LONG	Satellite element 6 is longitude of ascending node rather than right ascension.
18.	MATRIX	Run MATRIX postprocessor
	pstart,plen,numper,acfrac	pstart = time step to start MATRIX output plen = period length, in time steps numper = number of periods acfrac = minimum acceptable fraction of period that satellite must see target to count as a hit. MATRIX will print out a graphic representation of the sightings of the satellites on each target (see sample output).

3.0 SAMPLE INPUT AND OUTPUT

3.1 Descriptions of Sample Problems

This section demonstrates the process a user can follow to use LORI to calculate the statistics he needs. Suppose a user has the following two problems:

1. Given two systems of two satellites each, what is the range of delays between the first system sighting a target and a subsequent sighting by the second system of the same target? The first satellite system requires seeing a ground station at the same time it sees the target. The satellites in the first system have an antenna footprint determined by an elevation angle of 10°; that is, they must have an elevation angle of 10° above the horizon before they can see a target. The first system has a separate antenna for communication with the ground station, and its footprint is determined by the station masks of the ground stations, which in this case are all the same; elevation angles of 5° (the radio horizon). The satellites in the second system have an antenna with a co-dip angle of 40°. The user also wants the range of delays for the second sighting of the second system after the first sighting of the first system. If a delay is less than 7 min it is not to count (for instance, a delay that short is too small for command, communication, and control processing to program the second system), and a delay of more than 7 h is too long to count. In addition, the user is interested in the statistics (number and duration) on simultaneous sightings of the same target by both systems. And finally, the user is suspicious of the output and wants to look at the explicit sighting history of each satellite vs each target.

2. Given a satellite system of four satellites, what are the delays between sightings of four different targets for the total system. The satellites do not need to see a ground station. The satellites are in the same plane and are phased equally in that plane, that is, they are 90° apart in true anomaly.

The first two targets are moving on a great circle, in opposite directions, at a speed of 30 knots. The other two targets are stationary and correspond to the endpoints of the great circle. The output for the targets is to be on one file. The satellite's antenna footprints are the 5° radio horizon; that is, the satellite must be 5° above the horizon as seen from the target to register as a sighting. The run is to be for four days, with a 1-min step size. The user is given osculating elements for the satellites which include the longitude of the ascending node. In addition, the user wants a graphic output of the coverage of the four satellites as a function of time.

3.2 Program Setup

For these runs the maximum number of satellites is four, and the maximum number of targets is four. The user could make sure the program could handle these numbers by editing the source by hand and changing the parameter statements in every routine, then recompiling, relinking, and then running the same task for both sets of runs. The program REDIM will do most of this automatically, however. REDIM will prompt the user for the types of runs he expects to make and then create command files that will edit the source files, recompile, relink, and purge the directory of intermediate files. The user can then run LORI with no further changes. The sample output shows the user running REDIM, REDIM'S prompts, the user's answers, and the user running LORI. The following two sections show the command files B.COM and C.COM created by the program REDIM. After running REDIM the user invokes these by merely typing "@B". Note that REDIM will give the user the maximum recommended value for NUMHITS, given the other parameters of the run. This value is for running on core-limited machines and is a recommendation only. For the actual run, the user can input any value he chooses, as the sample shows.

3.3 Input and Output

The input and output for the two runs of LORI follow the command file listings. The input decks for the two runs precede their respective output listings and include comments to allow comparison to the problem description.

The first parts of each output from LORI are similar and merely repeat the input decks in more readable form. This section is repeated, the second section missing the information on the ground stations and satellites. If this information is sensitive, the first page of the output can be ripped off, leaving the output with enough information to identify the specific run to the user, but without sensitive information on it. If the output is sent to separate files for each target, each file will have essentially the same header. If the output from several targets is sent to one file, as it is in case 1, only one header appears.

Case 1

The first section of output after the headers is the delay section output. The information printed is sectioned by target. The times listed are in minutes. Each line includes

1. The target number
2. The start of the satellite system 1 sighting
3. The start of the satellite system 2 sighting
4. The delay between the two sightings
5. The system 1 satellite involved
6. The system 2 satellite involved
7. The ground station the system 1 satellite saw
8. The ground station the system 2 satellite saw.

If the satellite did not require a simultaneous ground station sighting, the number listed will be zero.

If the user requested more than one revisit of a system 2 sighting to a system 1 sighting, the subsequent revisits will be listed together with the first. The multiple revisits are separated by single blank lines from the other groups. Changes in days are marked by three blank lines. The delays listed will be the delay between the start of the sighting by the system 1 satellite to the start of the sighting by the system 2 satellite, with the following exceptions:

1. If the satellite is required to see a ground station, the start time will be the first time it sees both.

2. If the delay will be less than the minimum delay specified, the system 2 time is moved forward enough to meet the minimum delay requirement, as long as the system 2 satellite can still see the target at that time.

3. If the delay will be greater than the maximum delay specified, the system 1 start time will be moved forward enough to meet the criterion. The last two cases are easy to detect when they occur, as the delay listed will be one of the two limits.

The debug output follows the section on delays. If sent to a single file, the delay output for each target will be listed first, then the debug output for each target, then the simultaneous coverage output. As SIMUL destroys the internal table, it is called last. The debug output is in integer time steps. The output is essentially a listing of the two internal tables, GS and IGS. The entries are listed by satellite in columns. There are three numbers per sighting: the time step the satellite first saw the target, the time step it next did not see the target, and the ground station for that sighting. The search for a ground station quits as soon as it finds one, so if more than one ground station is visible, the program will list only the lowest indexed one (i.e., the earliest one on the input list).

The simultaneous coverage output follows the debug printout. There are several options on this output, but in its longest form it lists the following:

1. The target

2. The start time, stop time, and duration of each individual simultaneous event.

3. The sum of the simultaneous coverage between the two systems

4. The total merged coverage of each system

5. The total coverage of each satellite in both systems

6. The start and stop time for the two systems (as if each system were one satellite). These are the merged times of each system.

7. The start and stop time of both systems merged into one, with statistics on duration and delay of all merged sightings.

The second and last cases are optional and can be omitted by specifying the proper keyword.

Case 2

For the second case the delays between sightings by a single satellite system of four satellites is desired. The postprocessor DELAY works only for two satellite systems, so the routine SIMUL is used. SIMUL merges the individual sightings of each satellite in each system into two system sighting tables. For each system, if the start-stop pair of times for one satellite overlap that of another satellite, the times are merged into one start-stop pair. What results is two lists of start-stop times, one for each system. SIMUL can work just as well, however, on one system as on two. The output will show no simultaneous coverage because the second system doesn't exist, but what the user is interested in is the output that is printed due to the keyword TOTAL. Both TOTAL and SYSTMCOV list start-stop times for merged systems: TOTAL merges all satellites into one sighting history; SYSTMCOV merges the sightings into two sets, one for system 1, and one for system 2. If there is only one system (NSAT 2 = 0), both TOTAL and SYSTMCOV will list the same start-stop times. TOTAL, however, will list the duration and delay between the sightings, as well as the final statistics for the run. With SHORTOUT only the statistics will be output.

Note that the first two targets are moving on great circles and that they reach their end points before the end of the run. The program makes note of the fact on the output and quits processing these two targets, while continuing to the end of the run for the rest of the targets (stationary).

For the graphics output, the user specifies the MATRIX postprocessor, with the specified time step to start the periods, the length of the periods, the number of periods, and the minimum acceptable fraction of the period the satellite must see the target to register as a sighting for that period.

The output from MATRIX follows, with the time period as the row and the satellite as the column. The output includes the sums of the rows and the columns. A one (1) means that the satellite in that column saw the target for at least the fraction of the period specified for the period in the row. A blank means it did not. Note that the output from MATRIX can be misleading for certain selections of period resolution, acceptable fraction, and the duration of the satellite sighting. For instance, if the sighting duration is 20 min, the period selected is 1 h, and the fraction is 50%, no hits will register at all. Also, if the sighting straddles two periods, it is possible that it will not show up in either period, as the fraction in each period might be less than the minimum, even though the total duration would be long enough if it were in one period only.

The simultaneous coverage output is listed next. Each target is identified by latitude and longitude. For this type of run (no system 2 satellites), we see that, as expected, the simultaneous coverage time is zero, as is the total time of coverage for the second system. From the total coverage of system 1 and the individual coverage for each satellite the user can determine how much overlapping of coverage occurred within the system (if the individual totals equal the system total, there was no overlap). Finally, the start-stop times of the system as a whole are listed. From this the user can determine the delays in coverage. If the system sees the target during the entire length of the run with no gaps, all that will be listed is a system 1 start time of zero and a system 1 stop time of the length of the run. The run in this example used the keyword TOTAL. This tells SIMUL to merge all the hits of all the satellites, as if they were one system. This allows the use of DELAY, which requires two systems, in the same run with SIMUL, in those cases where one wants the merged times for both systems together in the same run with the delay output. For this particular case essentially the same output could have been obtained without the keyword TOTAL, but with the keyword SYSTMCOV.

3.4 Graphic Representation of Output

As can be seen, LORI can generate a lot of data in a single run. In general, a user will not be interested in the specific incidents in a run, but in the statistics generated by the entire run, i.e., maxima, minima, averages, means, and totals. Some of these statistics are printed out by the postprocessors, but they may not be the ones the user needs. In any case for analysis purposes a graphic display of the statistics will usually give more insight into what is occurring with a particular system. Samples of graphic outputs obtained by processing the output files from LORI follow (Figs. 5-8). Graphics routines for immediate display of output are not included in LORI for several reasons. The general nature of graphics may require several attempts before the desired scaling, labeling, and multiple curves are obtained. This is better suited to a mode of running whereby the data are generated by LORI and stored on output files, and these files are then processed by assorted graphics programs. On some machines graphics packages are relatively large and do not permit much computing in the same program as the plotting is done. Graphics packages vary widely from machine to machine and from installation to installation. For this reason the programs used to generate the plots shown are not listed.

Sample Run of REDIM and LORI

```
run redim
INPUT MAXIMUM # OF  SATS, MAXIMUM # OF TARGETS,
AND MAXIMUM # OF GROUND STATIONS.
  I WILL MAXIMIZE FOR LENGTH OF RUN .
  IF THE VALUES ARE NOT WHAT YOU WANT YOU MAY INPUT YOUR OWN
4,4,4
OKAY, WE ARE SET TO RUN WITH THE FOLLOWING DIMENSIONS
# SATS =    4 # TARGETS =    4 # G. STA. =    4
  # HITS =   862
IF YOU WANT TO CONTINUE TYPE A 0,ELSE A 1
0
TYPE IN # OF SATS, # OF TARGETS, # OF STATIONS
 AND # OF HITS
4,4,4,800
$ @b
$ run lori
TYPE IN INPUT FILE NAME
input.one
TYPE IN HEADER TITLE
 test case one for writeup
TIMES IN SECONDS      PAGE      DIRECT     BUFFERED
    CPU   ELAPSED    FAULTS      I/O          I/O
   25.0     39.4         67          14           17

$ run lori
TYPE IN INPUT FILE NAME
input.two
TYPE IN HEADER TITLE
 test case two for writeup
TIMES IN SECONDS      PAGE      DIRECT     BUFFERED
    CPU   ELAPSED    FAULTS      I/O          I/O
   36.0     52.7         70          33           33
```

Command File B. COM

```
$EDIT/SLP LORI.FOR        /NOAUDIT_TRAIL
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
/
$EDIT/SLP INPUT.FOR       /NOAUDIT_TRAIL
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
/
$EDIT/SLP SETUP.FOR       /NOAUDIT_TRAIL
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
/
$EDIT/SLP COREL8.FOR      /NOAUDIT_TRAIL
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
/
$EDIT/SLP DELAY.FOR       /NOAUDIT_TRAIL
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
/
$EDIT/SLP SIMUL.FOR       /NOAUDIT_TRAIL
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
/
$EDIT/SLP MATRIX.FOR      /NOAUDIT_TRAIL
-/PARAMETER MAX/,.
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
/
$PURGE LORI.FOR
$PURGE INPUT.FOR
$PURGE SETUP.FOR
$PURGE COREL8.FOR
$PURGE DELAY.FOR
$PURGE SIMUL.FOR
$PURGE MATRIX.FOR
$@C
```

Command File C. COM

```
$FORTRAN LORI.FOR
$FORTRAN INPUT.FOR
$FORTRAN SETUP.FOR
$FORTRAN COREL8.FOR
$FORTRAN DELAY.FOR
$FORTRAN SIMUL.FOR
$FORTRAN MATRIX.FOR
$LINK/MAP:LORI LORI,INPUT,SETUP,COREL8,DELAY,SIMUL-
 ,MATRIX,[UTILITY]VAXTIME
$DELETE LORI.OBJ;*
$DELETE INPUT.OBJ;*
$DELETE SETUP.OBJ;*
$DELETE COREL8.OBJ;*
$DELETE DELAY.OBJ;*
$DELETE SIMUL.OBJ;*
$DELETE MATRIX.OBJ;*
$PURGE LORI.EXE,LORI.MAP
$PURGE B.COM,C.COM
```

Sample Input File INPUT.ONE

```
1.                    Time step in minutes (dt)
80 1 1 0 0 0          Run start time (yy mm dd hh mm sss.)
80 1 4 0 0 0          Run stop time (yy mm dd hh mm sss.)
2                     Number of sys1 satellites (nsat1)
2                     Number of sys2 satellites (nsat2)
4                     Number of ground stations (ngs)
30 270 0. 5.          Ground station elements (lat,lon,rad,elev. angle)
30 255 0  5.
45 105 0  5.
-45 60 0  5.         Satellite elements (a,e,i,m,w,o,epoch,ifgs,ant.type)
7489. .001 63. 230. 100. 120.   50 1 1 0 0 0.   1 1
10.0                                                ant. parameter
7489. .001 63.  50. 100. 120.   50 1 1 0 0 0.   1 1
10.0
7489. .001 63.   0. 115. 180.   50 1 1 0 0 0.   0 2
40.0
7489. .001 63. 180. 115. 180.   50 1 1 0 0 0.   0 2
40.0
DELAY                 Calculate delays from sys1 to sys2 sightings
2                     Calculate two revisits of sys2 to a sys1 sighting(inc)
7                     Minimum revisit delay (tmin)
420                   Maximum revisit delay (tmax)
DEBUGOUT              Print out internal GS and IGS files
TARINPUT              Targets are input and follow (lat,lon,radius)
1                     Number of stationary targets input
60 245 0
SIMUL                 Calculate simultaneous coverage of sys1 and sys2
SYSTMCOV              Print out coverage times for the systems
START                 End of input, start run
```

Sample Output of Input File INPUT.ONE

```
test case one for writeup
   STEPSIZE(MIN)=    1.00
# SYS1 SATS =   2  # SYS2 SATS =   2  # TARGETS =   1  # GROUND STATIONS =   4

              YR  MO  DAY  HR  MIN     SEC
START TIME IS 80   1   1    0   0      0.00
STOP TIME IS  80   1   4    0   0      0.00

MINIMUM DELAY(MIN)=      7.0  MAX DELAY(MIN)=    420.0
NUMBER OF SYS2 SAT REVISITS PER SYS1 SAT VISIT =   2


              TARGET        [DEG]        [DEG]        [DEG]
LOCATION      TYPE          LAT.         LONG.        RADIUS
TARGET #   1  UNMOVING      60.00000     245.00000    0.00000

G. STATION                  LAT.         LONG.        RADIUS      ELEVATION
G. STA. #  1                30.00000     270.00000    0.00000     5.00000
G. STA. #  2                30.00000     255.00000    0.00000     5.00000
G. STA. #  3                45.00000     105.00000    0.00000     5.00000
G. STA. #  4               -45.00000      60.00000    0.00000     5.00000

SATELLITE ELEMENTS       [DEG]     [DEG]    [DEG]    [DEG]                                          [DEG]     [DEG]
                                   MEAN                                        IF   ANT.
  #   A(KM)        E        INC      ANOM     PERI     NODE    YR  M  D  H  M  SEC GS   TYPE      PARAM1    PARAM2

  1   7489.0000 0.00100 63.0000 230.000 100.000 120.000 50  1  1  0  0  0.00 T   ELEV      10.000     0.000
  2   7489.0000 0.00100 63.0000  50.000 100.000 120.000 50  1  1  0  0  0.00 T   ELEV      10.000     0.000
  1   7489.0000 0.00100 63.0000   0.000 115.000 180.000 50  1  1  0  0  0.00 F   CORP      40.000     0.000
  2   7489.0000 0.00100 63.0000 180.000 115.000 180.000 50  1  1  0  0  0.00 F   CORP      40.000     0.000
```

GP	SYS1 TIME	SYS2 TIME	DELAY	SYS1SAT	SYS2SAT	SYS1GS	SYS2GS
1	216.0	267.0	51.0	1.	1.	1.	0.
1	216.0	324.0	108.0	1.	2.	1.	0.
1	271.0	324.0	53.0	2.	2.	1.	0.
1	271.0	380.0	109.0	2.	1.	1.	0.
1	326.0	380.0	54.0	1.	1.	2.	0.
1	326.0	436.0	110.0	1.	2.	2.	0.
1	381.0	436.0	55.0	2.	2.	2.	0.
1	381.0	492.0	111.0	2.	1.	2.	0.
1	436.0	492.0	56.0	1.	1.	2.	0.
1	436.0	547.0	111.0	1.	2.	2.	0.
1	492.0	547.0	55.0	2.	2.	2.	0.
1	492.0	602.0	110.0	2.	1.	2.	0.
1	1275.0	1665.0	390.0	1.	1.	1.	0.
1	1327.0	1665.0	338.0	2.	1.	1.	0.
1	1327.0	1720.0	393.0	2.	2.	1.	0.
1	1380.0	1665.0	285.0	1.	1.	2.	0.
1	1380.0	1720.0	340.0	1.	2.	2.	0.
1	1435.0	1665.0	230.0	2.	1.	2.	0.
1	1435.0	1720.0	285.0	2.	2.	2.	0.
1	1668.0	1720.0	52.0	2.	2.	1.	0.
1	1668.0	1776.0	108.0	2.	1.	1.	0.
1	1723.0	1776.0	53.0	1.	1.	2.	0.
1	1723.0	1831.0	108.0	1.	2.	2.	0.
1	1777.0	1831.0	54.0	2.	2.	2.	0.
1	1777.0	1887.0	110.0	2.	1.	2.	0.
1	1831.0	1887.0	56.0	1.	1.	2.	0.
1	1831.0	1943.0	112.0	1.	2.	2.	0.
1	1887.0	1943.0	56.0	2.	2.	2.	0.
1	1887.0	1998.0	111.0	2.	1.	2.	0.
1	2724.0	3062.0	338.0	2.	1.	1.	0.
1	2724.0	3117.0	393.0	2.	2.	1.	0.

1	2778.Ø	3Ø62.Ø	284.Ø	1.	1.	1.	Ø.
1	2778.Ø	3117.Ø	339.Ø	1.	2.	1.	Ø.
1	2831.Ø	3Ø62.Ø	231.Ø	2.	1.	2.	Ø.
1	2831.Ø	3117.Ø	286.Ø	2.	2.	2.	Ø.
1	3Ø67.Ø	3117.Ø	5Ø.Ø	2.	2.	1.	Ø.
1	3Ø67.Ø	3172.Ø	1Ø5.Ø	2.	1.	1.	Ø.
1	3121.Ø	3172.Ø	51.Ø	1.	1.	1.	Ø.
1	3121.Ø	3228.Ø	1Ø7.Ø	1.	2.	1.	Ø.
1	3174.Ø	3228.Ø	54.Ø	2.	2.	2.	Ø.
1	3174.Ø	3283.Ø	1Ø9.Ø	2.	1.	2.	Ø.
1	3228.Ø	3283.Ø	55.Ø	1.	1.	2.	Ø.
1	3228.Ø	334Ø.Ø	112.Ø	1.	2.	2.	Ø.
1	3283.Ø	334Ø.Ø	57.Ø	2.	2.	2.	Ø.
1	3283.Ø	3395.Ø	112.Ø	2.	1.	2.	Ø.
1	334Ø.Ø	3395.Ø	55.Ø	1.	1.	2.	Ø.
1	334Ø.Ø	3451.Ø	111.Ø	1.	2.	2.	Ø.

```
FOR ALL DELAYS BETWEEN SYSTEMS
MAXIMUM DELAY =     393.ØØ
MINIMUM DELAY =      5Ø.ØØ
AVERAGE DELAY =     147.Ø9
TOTAL NUMBER OF DELAYS =       47
FOR THE   1  DELAY BETWEEN SYSTEMS
MAXIMUM DELAY =     39Ø.ØØ
MINIMUM DELAY =      5Ø.ØØ
AVERAGE DELAY =     125.54
TOTAL NUMBER OF DELAYS =       24
FOR THE   2  DELAY BETWEEN SYSTEMS
MAXIMUM DELAY =     393.ØØ
MINIMUM DELAY =     1Ø5.ØØ
AVERAGE DELAY =     169.57
TOTAL NUMBER OF DELAYS =       23
```

```
DEBUG OUTPUT FOR TARGET #  1
START OF SIGHTING, START OF NON-SIGHTING, AND STATION #
     SAT #  1  SAT #  2  SAT #  3  SAT #  4
       216        271        267        213
       220        275        272        216
         1          1          0          0
       326        381        380        324
       332        387        385        330
         2          2          0          0
       436        492        492        436
       442        495        497        442
         2          2          0          0
      1275       1327        602        547
      1280       1334        607        552
         1          1          0          0
      1380       1435       1665       1720
      1388       1441       1669       1726
         2          2          0          0
      1723       1668       1776       1831
      1728       1672       1781       1837
         2          1          0          0
      1831       1777       1887       1943
      1838       1783       1892       1949
         2          2          0          0
      2778       1887       1998       2054
      2785       1891       2003       2058
         1          2          0          0
      3121       2724       3062       3117
      3124       2732       3066       3122
         1          1          0          0
      3228       2831       3172       3228
      3234       2838       3178       3233
         2          2          0          0
      3340       3067       3283       3340
      3343       3068       3288       3345
         2          1          0          0
      4175       3174       3395       3451
      4182       3180       3402       3456
         1          2          0          0
      4285       3283 2147483647 2147483647
      4291       3288 2147483647 2147483647
         2          2          0          0
2147483647       4123          0          0
2147483647       4128          0          0
         0          1          0          0
         0       4228          0          0
         0       4236          0          0
         0          2          0          0
         0 2147483647          0          0
         0 2147483647          0          0
         0          0          0          0
```

SIMULTANEOUS COVERAGE FOR TARGET 60.00 245.00
TOTAL COVERAGE FOR EACH SATELLITE IS
73.00 80.00 61.00 62.00

START OF COVERAGE	END OF MUTUAL COVERAGE	PERIOD OF COVERAGE(IN MIN.)
271.0	272.0	1.0
326.0	330.0	4.0
381.0	385.0	4.0
436.0	442.0	6.0
492.0	495.0	3.0
1668.0	1669.0	1.0
1723.0	1726.0	3.0
1777.0	1781.0	4.0
1831.0	1837.0	6.0
1887.0	1891.0	4.0
3121.0	3122.0	1.0
3174.0	3178.0	4.0
3228.0	3233.0	5.0
3283.0	3288.0	5.0
3340.0	3343.0	3.0

TOTAL TIME OF SIMULTANEOUS COVERAGE IS 54.00 MIN.
TOTAL COVERAGE FOR SYS1 AND SYS2 IS 153.00 123.00

SYS1STRT	SYS1STP	SYS2STRT	SYS2STP
216.00	220.00	213.00	216.00
271.00	275.00	267.00	272.00
326.00	332.00	324.00	330.00
381.00	387.00	380.00	385.00
436.00	442.00	436.00	442.00
492.00	495.00	492.00	497.00
1275.00	1280.00	547.00	552.00
1327.00	1334.00	602.00	607.00
1380.00	1388.00	1665.00	1669.00
1435.00	1441.00	1720.00	1726.00
1668.00	1672.00	1776.00	1781.00
1723.00	1728.00	1831.00	1837.00
1777.00	1783.00	1887.00	1892.00
1831.00	1838.00	1943.00	1949.00
1887.00	1891.00	1998.00	2003.00
2724.00	2732.00	2054.00	2058.00
2778.00	2785.00	3062.00	3066.00
2831.00	2838.00	3117.00	3122.00
3067.00	3068.00	3172.00	3178.00
3121.00	3124.00	3228.00	3233.00
3174.00	3180.00	3283.00	3288.00
3228.00	3234.00	3340.00	3345.00
3283.00	3288.00	3395.00	3402.00
3340.00	3343.00	3451.00	3456.00
4123.00	4128.00	0.00	0.00
4175.00	4182.00	0.00	0.00
4228.00	4236.00	0.00	0.00
4285.00	4291.00	0.00	0.00
0.00	0.00	0.00	0.00

Sample Input File INPUT.TWO

```
1.                       Time step in minutes (dt)
80 1 1 0 0 0             Run start time (yy mm dd hh mm sss.)
80 1 5 0 0 0             Run stop time (yy mm dd hh mm sss.)
1                        Number of sys1 satellites (nsat1)(will be reset below)
0                        Number of sys2 satellites (nsat2)
0                        Number of ground stations (ngs)
7489 .001 63 230 100 120 50 1 1 0 0 0 0 1      Satellite parameters
5.                                              Antenna parameter
LONG                     Satellite elements use longitude, not R. A.
SIMUL                    Calculate simultaneous coverage of sys1 and sys2
TOTAL                    Print out merged sightings
SHPTRK                   Targets that follow are moving, along great circles
2                        Number of moving targets
0. 0. 30. 30. 30.        Start lat and long, end lat and long,
30. 30. 0. 0. 30.        and speed in knots.
TARINPUT                 Target elements are input
2                        Number of non-moving targets
0. 0. 0.                 latitude, longitude, radius(deg.)
30. 30. 0.
VARIATIO                 Satellite elements are a grid
4 0. 90. 4               (true anomaly from 0 to 360 in 90 deg steps)
MATRIX
1 30 48 .10
SINGLFIL                 All output is to go to file FOR008.DAT
TRUE                     Satellite input uses true anomaly
START                    End of input, start run
```

Sample Output of Input File INPUT.TWO

```
test case two for writeup
   STEPSIZE(MIN)=    1.00
# SYS1 SATS =   4  # SYS2 SATS =   0  # TARGETS =   4  # GROUND STATIONS =   0

                YR  MO  DAY  HR  MIN     SEC
START TIME IS  80   1    1   0    0      0.00
STOP TIME IS   80   1    5   0    0      0.00


                  TARGET       [DEG]        [DEG]        [DEG]
LOCATION          TYPE         LAT.         LONG.        RADIUS
TARGET #   1  SHIPTRAK      0.00000      0.00000      0.00000 TO      30.00000      30.00000 AT  30.00000 KNOTS
TARGET #   2  SHIPTRAK     30.00000     30.00000      0.00000 TO       0.00000       0.00000 AT  30.00000 KNOTS
TARGET #   3  UNMOVING      0.00000      0.00000      0.00000
TARGET #   4  UNMOVING     30.00000     30.00000      0.00000

MATRIX GRID STARTS AT STEP      1
IS      30 STEPS LONG PER PERIOD,
FOR     48 PERIODS,
AND WITH AN ACCEPTABLE FRACTION OF  10.00 PERCENT

VARIATION OF SATELLITE ELEMENT  4 FROM    0.000 BY   90.000 ,     4 TIMES
SATELLITE ELEMENTS       [DEG]     [DEG]    [DEG]    [DEG]                                      [DEG]      [DEG]
                                   TRUE              LONG                            IF   ANT.
  #   A(KM)         E      INC     ANOM     PERI     NODE    YR  M  D  H  M  SEC GS   TYPE     PARAM1     PARAM2

  1   7489.0000 0.00100 63.0000    0.000 100.000 120.000 50  1  1  0  0  0.00 F    ELEV      5.000      0.000
```

Sample Output of Input File INPUT.TWO

```
test case two for writeup
   STEPSIZE(MIN)=    1.00
# SYS1 SATS =   4  # SYS2 SATS =   0  # TARGETS =   4  # GROUND STATIONS =   0

              YR  MO  DAY  HR  MIN     SEC
START TIME IS 80   1   1    0   0      0.00
STOP TIME IS  80   1   5    0   0      0.00


               TARGET       [DEG]        [DEG]        [DEG]
LOCATION        TYPE         LAT.         LONG.        RADIUS
TARGET #   1  SHIPTRAK     0.00000      0.00000      0.00000 TO      30.00000      30.00000 AT  30.00000 KNOTS
TARGET #   2  SHIPTRAK    30.00000     30.00000      0.00000 TO       0.00000       0.00000 AT  30.00000 KNOTS
TARGET #   3  UNMOVING     0.00000      0.00000      0.00000
TARGET #   4  UNMOVING    30.00000     30.00000      0.00000

MATRIX GRID STARTS AT STEP      1
IS      30 STEPS LONG PER PERIOD,
FOR     48 PERIODS,
AND WITH AN ACCEPTABLE FRACTION OF  10.00 PERCENT
TARGET #    1 STOPPED AT TIME STEP         4971
 AT TIME =    298260.00 SEC
TARGET #    2 STOPPED AT TIME STEP         4971
 AT TIME =    298260.00 SEC
```

HIT MATRIX FOR TARGET # 1 AT LAT, LONG = (0.00, 0.00)

PER #	SAT. #				
	1	2	3	4	TOT.# SATS
1					0
2					0
3					0
4					0
5					0
6					0
7					0
8					0
9					0
10					0
11					0
12			1		1
13		1			1
14	1				1
15				1	1
16			1		1
17		1			1
18	1				1
19				1	1
20					0
21					0
22					0
23					0
24					0
25					0
26					0
27					0
28					0
29					0
30					0
31					0
32					0
33					0
34					0
35					0
36					0
37	1	1			2
38	1			1	2
39			1	1	2
40		1	1		2
41	1	1			2
42				1	1
43			1		1
44					0
45					0
46					0
47					0
48					0
	5	5	5	5	

TOTAL NUMBER OF PERIODS PER SAT

HIT MATRIX FOR TARGET # 2 AT LAT, LONG = (30.00, 30.00)

PER #	SAT. # 1	2	3	4	TOT.# SATS
1					0
2					0
3					0
4					0
5			1		1
6		1			1
7	1			1	2
8			1	1	2
9		1	1		2
10	1	1			2
11	1			1	2
12			1		1
13		1			1
14	1				1
15				1	1
16					0
17					0
18					0
19					0
20					0
21					0
22					0
23					0
24					0
25					0
26					0
27					0
28					0
29					0
30					0
31					0
32					0
33					0
34					0
35				1	1
36			1		1
37		1			1
38	1				1
39			1	1	2
40		1	1		2
41	1	1			2
42	1			1	2
43			1	1	2
44					0
45					0
46					0
47					0
48					0
	7	7	8	8	

TOTAL NUMBER OF PERIODS PER SAT

HIT MATRIX FOR TARGET # 3 AT LAT, LONG = (0.00, 0.00)

PER #	SAT. # 1	2	3	4	TOT.# SATS
1					0
2					0
3					0
4					0
5					0
6					0
7					0
8					0
9					0
10					0
11					0
12					0
13		1			1
14	1				1
15				1	1
16			1		1
17		1			1
18	1				1
19				1	1
20			1		1
21					0
22					0
23					0
24					0
25					0
26					0
27					0
28					0
29					0
30					0
31					0
32					0
33					0
34					0
35					0
36					0
37	1	1			2
38	1			1	2
39			1	1	2
40		1	1		2
41	1				1
42				1	1
43			1		1
44		1			1
45					0
46					0
47					0
48					0
	5	5	5	5	

TOTAL NUMBER OF PERIODS PER SAT

HIT MATRIX FOR TARGET # 4 AT LAT, LONG = (30.00, 30.00)

PER #	SAT. # 1	2	3	4	TOT.# SATS
1					0
2					0
3					0
4					0
5			1		1
6		1			1
7	1			1	2
8			1	1	2
9		1	1		2
10	1	1			2
11	1			1	2
12			1		1
13		1			1
14	1				1
15					0
16					0
17					0
18					0
19					0
20					0
21					0
22					0
23					0
24					0
25					0
26					0
27					0
28					0
29					0
30					0
31					0
32					0
33					0
34	1				1
35				1	1
36			1		1
37		1			1
38	1				1
39				1	1
40			1		1
41	1	1			2
42	1			1	2
43			1	1	2
44			1		1
45					0
46					0
47					0
48					0
	8	6	8	7	

TOTAL NUMBER OF PERIODS PER SAT

SIMULTANEOUS COVERAGE FOR TARGET 0.00 0.00
TOTAL COVERAGE FOR EACH SATELLITE IS
219.00 222.00 197.00 207.00
TOTAL TIME OF SIMULTANEOUS COVERAGE IS 0.00 MIN.
TOTAL COVERAGE FOR SYS1 AND SYS2 IS 845.00 0.00
TOTAL MERGED COVERAGE FOR BOTH SYSTEMS

START	STOP	DURATION	DELAY
353.00	359.00	6.00	353.00
379.00	391.00	12.00	20.00
405.00	420.00	15.00	14.00
432.00	448.00	16.00	12.00
459.00	476.00	17.00	11.00
487.00	504.00	17.00	11.00
516.00	530.00	14.00	12.00
545.00	557.00	12.00	15.00
1080.00	1091.00	11.00	523.00
1107.00	1121.00	14.00	16.00
1134.00	1149.00	15.00	13.00
1160.00	1176.00	16.00	11.00
1188.00	1205.00	17.00	12.00
1217.00	1233.00	16.00	12.00
1246.00	1260.00	14.00	13.00
1276.00	1285.00	9.00	16.00
1720.00	1726.00	6.00	435.00
1745.00	1757.00	12.00	19.00
1773.00	1787.00	14.00	16.00
1799.00	1815.00	16.00	12.00
1826.00	1843.00	17.00	11.00
1854.00	1871.00	17.00	11.00
1882.00	1898.00	16.00	11.00
1910.00	1923.00	13.00	12.00
1940.00	1949.00	9.00	17.00
2482.00	2490.00	8.00	533.00
2509.00	2520.00	11.00	19.00
2534.00	2549.00	15.00	14.00
2561.00	2578.00	17.00	12.00
2589.00	2606.00	17.00	11.00
2616.00	2632.00	16.00	10.00
2645.00	2660.00	15.00	13.00
2674.00	2687.00	13.00	14.00
2705.00	2714.00	9.00	18.00
3087.00	3093.00	6.00	373.00
3112.00	3124.00	12.00	19.00
3139.00	3153.00	14.00	15.00
3167.00	3183.00	16.00	14.00
3194.00	3211.00	17.00	11.00
3221.00	3238.00	17.00	10.00
3249.00	3265.00	16.00	11.00
3278.00	3292.00	14.00	13.00
3307.00	3317.00	10.00	15.00
3884.00	3889.00	5.00	567.00
3908.00	3920.00	12.00	19.00
3934.00	3948.00	14.00	14.00

3960.00	3976.00	16.00	12.00
3988.00	4005.00	17.00	12.00
4016.00	4033.00	17.00	11.00
4045.00	4060.00	15.00	12.00
4074.00	4087.00	13.00	14.00
4103.00	4115.00	12.00	16.00
4133.00	4142.00	9.00	18.00
4453.00	4462.00	9.00	311.00
4480.00	4492.00	12.00	18.00
4507.00	4520.00	13.00	15.00
4521.00	4522.00	1.00	1.00
4533.00	4549.00	16.00	11.00
4562.00	4578.00	16.00	13.00
4590.00	4606.00	16.00	12.00
4618.00	4634.00	16.00	12.00
4645.00	4660.00	15.00	11.00
4675.00	4687.00	12.00	15.00
4705.00	4710.00	5.00	18.00

MAXIMUM TOTAL SIGHTING DURATION = 17.00
MINIMUM TOTAL SIGHTING DURATION = 1.00
AVERAGE TOTAL SIGHTING DURATION = 13.20
TOTAL NUMBER OF MERGED SIGHTINGS = 64
MAXIMUM TOTAL SIGHTING DELAY = 567.00
MINIMUM TOTAL SIGHTING DELAY = 1.00
AVERAGE TOTAL SIGHTING DELAY = 60.39
TOTAL TIME TARGET SEEN BY BOTH SYSTEMS = 845.00

SIMULTANEOUS COVERAGE FOR TARGET 30.00 30.00
TOTAL COVERAGE FOR EACH SATELLITE IS
222.00 226.00 221.00 221.00
TOTAL TIME OF SIMULTANEOUS COVERAGE IS 0.00 MIN.
TOTAL COVERAGE FOR SYS1 AND SYS2 IS 890.00 0.00
TOTAL MERGED COVERAGE FOR BOTH SYSTEMS

START	STOP	DURATION	DELAY
126.00	129.00	3.00	126.00
152.00	161.00	9.00	23.00
179.00	191.00	12.00	18.00
206.00	220.00	14.00	15.00
234.00	248.00	14.00	14.00
260.00	277.00	17.00	12.00
288.00	305.00	17.00	11.00
316.00	333.00	17.00	11.00
344.00	360.00	16.00	11.00
373.00	388.00	15.00	13.00
402.00	414.00	12.00	14.00
432.00	438.00	6.00	18.00
1034.00	1041.00	7.00	596.00
1058.00	1071.00	13.00	17.00
1086.00	1101.00	15.00	15.00
1112.00	1128.00	16.00	11.00
1139.00	1156.00	17.00	11.00
1167.00	1183.00	16.00	11.00

1196.00	1212.00	16.00	13.00
1224.00	1238.00	14.00	12.00
1253.00	1265.00	12.00	15.00
1283.00	1291.00	8.00	18.00
1662.00	1670.00	8.00	371.00
1688.00	1700.00	12.00	18.00
1715.00	1729.00	14.00	15.00
1742.00	1758.00	16.00	13.00
1771.00	1787.00	16.00	13.00
1797.00	1814.00	17.00	10.00
1826.00	1841.00	15.00	12.00
1854.00	1868.00	14.00	13.00
1883.00	1894.00	11.00	15.00
1912.00	1919.00	7.00	18.00
2482.00	2490.00	8.00	563.00
2509.00	2520.00	11.00	19.00
2534.00	2549.00	15.00	14.00
2561.00	2577.00	16.00	12.00
2589.00	2605.00	16.00	12.00
2616.00	2632.00	16.00	11.00
2644.00	2659.00	15.00	12.00
2673.00	2686.00	13.00	14.00
2704.00	2712.00	8.00	18.00
3144.00	3150.00	6.00	432.00
3170.00	3182.00	12.00	20.00
3197.00	3211.00	14.00	15.00
3223.00	3239.00	16.00	12.00
3250.00	3267.00	17.00	11.00
3278.00	3295.00	17.00	11.00
3306.00	3322.00	16.00	11.00
3334.00	3348.00	14.00	12.00
3363.00	3375.00	12.00	15.00
3930.00	3938.00	8.00	555.00
3955.00	3968.00	13.00	17.00
3982.00	3997.00	15.00	14.00
4008.00	4026.00	18.00	11.00
4036.00	4052.00	16.00	10.00
4064.00	4080.00	16.00	12.00
4094.00	4108.00	14.00	14.00
4123.00	4133.00	10.00	15.00
4154.00	4158.00	4.00	21.00
4626.00	4631.00	5.00	468.00
4651.00	4662.00	11.00	20.00
4678.00	4692.00	14.00	16.00
4705.00	4721.00	16.00	13.00
4732.00	4749.00	17.00	11.00
4759.00	4776.00	17.00	10.00
4789.00	4804.00	15.00	13.00
4817.00	4831.00	14.00	13.00
4847.00	4856.00	9.00	16.00

MAXIMUM TOTAL SIGHTING DURATION = 18.00
MINIMUM TOTAL SIGHTING DURATION = 3.00
AVERAGE TOTAL SIGHTING DURATION = 13.09
TOTAL NUMBER OF MERGED SIGHTINGS = 68

MAXIMUM TOTAL SIGHTING DELAY = 596.00
MINIMUM TOTAL SIGHTING DELAY = 10.00
AVERAGE TOTAL SIGHTING DELAY = 58.32
TOTAL TIME TARGET SEEN BY BOTH SYSTEMS = 890.00

SIMULTANEOUS COVERAGE FOR TARGET 0.00 0.00
TOTAL COVERAGE FOR EACH SATELLITE IS
199.00 203.00 197.00 196.00
TOTAL TIME OF SIMULTANEOUS COVERAGE IS 0.00 MIN.
TOTAL COVERAGE FOR SYS1 AND SYS2 IS 795.00 0.00
TOTAL MERGED COVERAGE FOR BOTH SYSTEMS

START	STOP	DURATION	DELAY
380.00	390.00	10.00	380.00
406.00	420.00	14.00	16.00
433.00	448.00	15.00	13.00
460.00	476.00	16.00	12.00
488.00	504.00	16.00	12.00
516.00	532.00	16.00	12.00
545.00	558.00	13.00	13.00
575.00	583.00	8.00	17.00
1079.00	1087.00	8.00	496.00
1105.00	1117.00	12.00	18.00
1131.00	1146.00	15.00	14.00
1157.00	1174.00	17.00	11.00
1185.00	1203.00	18.00	11.00
1214.00	1230.00	16.00	11.00
1242.00	1256.00	14.00	12.00
1271.00	1283.00	12.00	15.00
1302.00	1309.00	7.00	19.00
1804.00	1814.00	10.00	495.00
1830.00	1844.00	14.00	16.00
1856.00	1872.00	16.00	12.00
1884.00	1900.00	16.00	12.00
1911.00	1927.00	16.00	11.00
1939.00	1955.00	16.00	12.00
1967.00	1982.00	15.00	12.00
1998.00	2008.00	10.00	16.00
2506.00	2512.00	6.00	498.00
2530.00	2542.00	12.00	18.00
2556.00	2571.00	15.00	14.00
2582.00	2598.00	16.00	11.00
2610.00	2626.00	16.00	12.00
2638.00	2654.00	16.00	12.00
2666.00	2680.00	14.00	12.00
2695.00	2708.00	13.00	15.00
2726.00	2733.00	7.00	18.00
3229.00	3238.00	9.00	496.00
3254.00	3268.00	14.00	16.00
3280.00	3296.00	16.00	12.00
3308.00	3324.00	16.00	12.00
3335.00	3351.00	16.00	11.00
3363.00	3380.00	17.00	12.00

3393.00	3407.00	14.00	13.00
3423.00	3434.00	11.00	16.00
3928.00	3934.00	6.00	494.00
3953.00	3965.00	12.00	19.00
3980.00	3995.00	15.00	15.00
4006.00	4022.00	16.00	11.00
4034.00	4050.00	16.00	12.00
4062.00	4078.00	16.00	12.00
4091.00	4106.00	15.00	13.00
4119.00	4132.00	13.00	13.00
4150.00	4157.00	7.00	18.00
4653.00	4661.00	8.00	496.00
4679.00	4692.00	13.00	18.00
4705.00	4721.00	16.00	13.00
4733.00	4749.00	16.00	12.00
4759.00	4776.00	17.00	10.00
4789.00	4805.00	16.00	13.00
4817.00	4831.00	14.00	12.00
4847.00	4858.00	11.00	16.00

MAXIMUM TOTAL SIGHTING DURATION = 18.00
MINIMUM TOTAL SIGHTING DURATION = 6.00
AVERAGE TOTAL SIGHTING DURATION = 13.47
TOTAL NUMBER OF MERGED SIGHTINGS = 59
MAXIMUM TOTAL SIGHTING DELAY = 498.00
MINIMUM TOTAL SIGHTING DELAY = 10.00
AVERAGE TOTAL SIGHTING DELAY = 68.86
TOTAL TIME TARGET SEEN BY BOTH SYSTEMS = 795.00

SIMULTANEOUS COVERAGE FOR TARGET 30.00 30.00
TOTAL COVERAGE FOR EACH SATELLITE IS
250.00 254.00 255.00 251.00
TOTAL TIME OF SIMULTANEOUS COVERAGE IS 0.00 MIN.
TOTAL COVERAGE FOR SYS1 AND SYS2 IS 1010.00 0.00
TOTAL MERGED COVERAGE FOR BOTH SYSTEMS

START	STOP	DURATION	DELAY
125.00	130.00	5.00	125.00
151.00	162.00	11.00	21.00
179.00	192.00	13.00	17.00
206.00	221.00	15.00	14.00
232.00	248.00	16.00	11.00
260.00	277.00	17.00	12.00
288.00	304.00	16.00	11.00
316.00	333.00	17.00	12.00
344.00	359.00	15.00	11.00
374.00	387.00	13.00	15.00
403.00	412.00	9.00	16.00
1010.00	1015.00	5.00	598.00
1035.00	1045.00	10.00	20.00
1060.00	1074.00	14.00	15.00
1088.00	1103.00	15.00	14.00
1115.00	1131.00	16.00	12.00
1142.00	1158.00	16.00	11.00

1170.00	1186.00	16.00	12.00
1199.00	1215.00	16.00	13.00
1228.00	1242.00	14.00	13.00
1256.00	1268.00	12.00	14.00
1287.00	1295.00	8.00	19.00
1550.00	1555.00	5.00	255.00
1575.00	1585.00	10.00	20.00
1602.00	1615.00	13.00	17.00
1631.00	1645.00	14.00	16.00
1658.00	1674.00	16.00	13.00
1685.00	1701.00	16.00	11.00
1713.00	1729.00	16.00	12.00
1741.00	1757.00	16.00	12.00
1770.00	1784.00	14.00	13.00
1797.00	1811.00	14.00	13.00
1826.00	1837.00	11.00	15.00
2435.00	2438.00	3.00	598.00
2459.00	2469.00	10.00	21.00
2485.00	2500.00	15.00	16.00
2513.00	2528.00	15.00	13.00
2540.00	2556.00	16.00	12.00
2567.00	2583.00	16.00	11.00
2595.00	2611.00	16.00	12.00
2623.00	2639.00	16.00	12.00
2652.00	2666.00	14.00	13.00
2680.00	2692.00	12.00	14.00
2711.00	2720.00	9.00	19.00
2743.00	2746.00	3.00	23.00
2975.00	2979.00	4.00	229.00
3000.00	3010.00	10.00	21.00
3027.00	3040.00	13.00	17.00
3055.00	3069.00	14.00	15.00
3082.00	3098.00	16.00	13.00
3109.00	3125.00	16.00	11.00
3137.00	3153.00	16.00	12.00
3166.00	3182.00	16.00	13.00
3194.00	3208.00	14.00	12.00
3221.00	3235.00	14.00	13.00
3250.00	3261.00	11.00	15.00
3860.00	3863.00	3.00	599.00
3884.00	3894.00	10.00	21.00
3909.00	3923.00	14.00	15.00
3936.00	3951.00	15.00	13.00
3963.00	3979.00	16.00	12.00
3991.00	4007.00	16.00	12.00
4019.00	4035.00	16.00	12.00
4047.00	4063.00	16.00	12.00
4076.00	4090.00	14.00	13.00
4106.00	4117.00	11.00	16.00
4135.00	4144.00	9.00	18.00
4167.00	4170.00	3.00	23.00
4399.00	4403.00	4.00	229.00
4424.00	4434.00	10.00	21.00
4451.00	4464.00	13.00	17.00

4479.00	4493.00	14.00	15.00
4506.00	4522.00	16.00	13.00
4533.00	4549.00	16.00	11.00
4562.00	4578.00	16.00	13.00
4590.00	4606.00	16.00	12.00
4618.00	4632.00	14.00	12.00
4645.00	4659.00	14.00	13.00
4675.00	4686.00	11.00	16.00

MAXIMUM TOTAL SIGHTING DURATION = 17.00
MINIMUM TOTAL SIGHTING DURATION = 3.00
AVERAGE TOTAL SIGHTING DURATION = 12.78
TOTAL NUMBER OF MERGED SIGHTINGS = 79
MAXIMUM TOTAL SIGHTING DELAY = 599.00
MINIMUM TOTAL SIGHTING DELAY = 11.00
AVERAGE TOTAL SIGHTING DELAY = 46.53
TOTAL TIME TARGET SEEN BY BOTH SYSTEMS = 1010.00

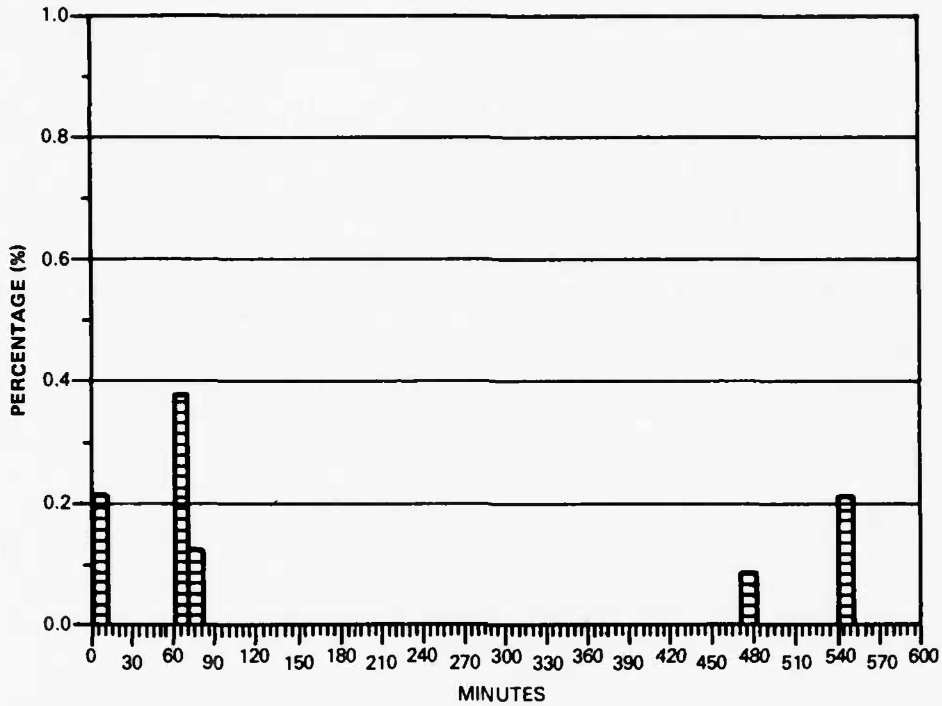

Fig. 5 — Distribution of delays between two systems; length of delay vs percentage of total number of delays

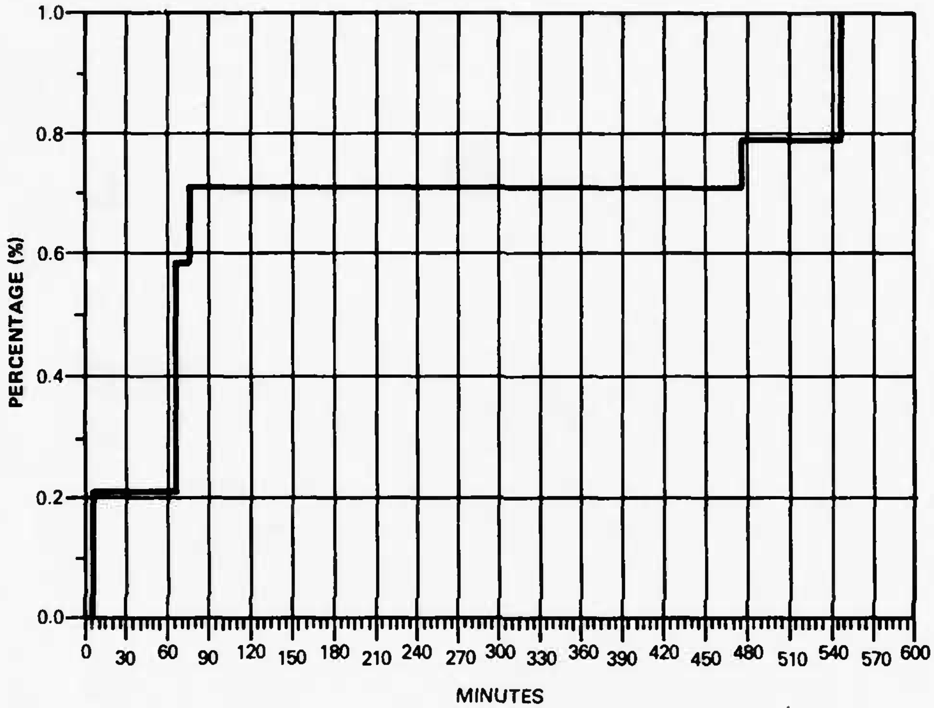

Fig. 6 — Cumulative distribution of delays between two systems; length of delay vs percentage of total number of delays

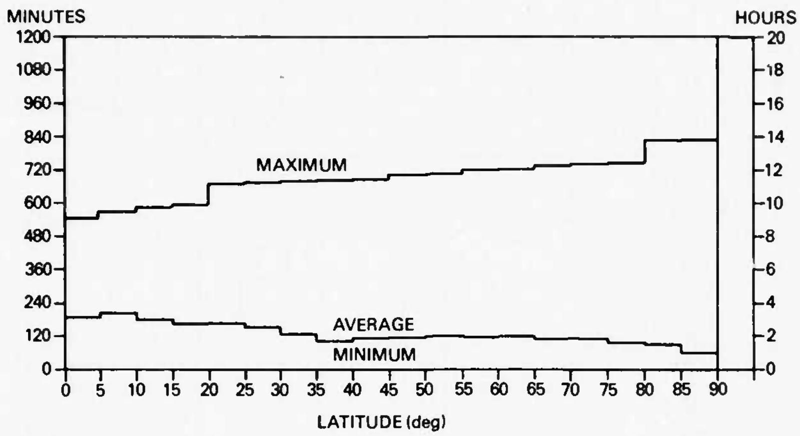

Fig. 7 — Sample of maximum, average, and minimum delays between sightings vs latitude (single system of four satellites)

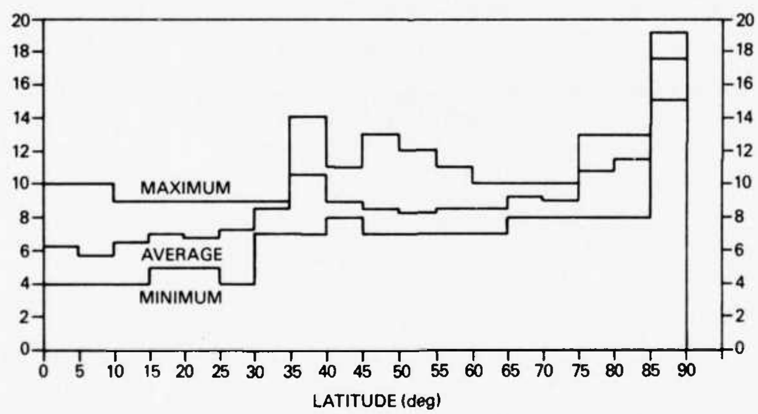

Fig. 8 — Sample of maximum, average, and minimum number of sightings per day vs latitude (single system of four satellites)

4.0 COMMON BLOCKS AND VARIABLES

Most parameter passing in LORI is done through common blocks. The main reason for this is the block structure of the program for overlapping on core-limited machines, and the multipass run mode of operation. In addition, it simplifies changes to the program, whether it is used with an "include" mode or with the REDIM preprocessor. For these reasons the input to and output from each routine are not specified in the routines, but are listed here.

/TRUTH/

SIMULT	True = call SIMUL
SINGL	True = output to single file FOR008.DAT
SHORT	True = short form of output for SIMUL
GRID	True = some targets are a grid
DBUG	True = print out GS file
DELA	True = call DELAY postprocessor
SYSTM	True = print out system times in SIMUL
NUMHIT	True = print out total number of hits by each satellite for each target
VARIA	True = satellites are a grid with one element changing
IFDONE(MAXGP)	True = target has reached end point
ALLDUN	True = all targets have reached end point
MAT	True = call MATRIX
TOTAL	True = merge system one and system two for printout of sightings in SIMUL (can be called with SYSTMCOV)
MEAN	True = satellite input uses mean anomaly False = satellite input uses true anomaly
SFIRST(MAXSAT)	True = first time satellite has seen target (for side-looking antenna type)

/NUM/

NSAT1	Number of system 1 satellites
NSAT2	Number of system 2 satellites
NGP	Number of targets
NGS	Number of ground stations
N1P1	NSAT1+1
NTOT	NSAT1+NSAT2
NGSP1	NGS+1
NGTOT	NGS+NGP

/FLAG/

IFLAG(MAXSAT,MAXGP)	Flag from ORBIT to COREL8 0 = no hit, 1 = hit on last time step
KMAX	Size of array OUTPUT (KMAX,5)
INC	Counter for DELAY: number of revisits
IFHTGP(MAXGP)	Flag for hit on target at current time step
NUMGP	Flag for current target
LSTART(MAXGP)	Flag for next location in OUTPUT
INP	Flag for start to INPUT and SETUP
NUMGS(MAXSAT,MAXGP)	Contains the number of the ground station in view for the current hit.

/CONS/	
DTR	Degrees to radians (pi/180)
RTD	Radians to degrees (180/pi)
PI	
TPI	2*pi
RE	Radius of earth in kilometers
XMU	μ(G*M)
XJ2	J2
WE	Rotation rate of the earth in rad/sec
LARGE	Largest integer available to machine
/TIME/	
TIME	Time in seconds since start of run
TSTOP	Length of run in seconds
DT	Time step in seconds
TMIN	Minimum delay from system 2 to system 1
TMAX	Maximum delay from system 2 to system 1
ISTOP(MAXGP)	Time, in time steps, that run ended for a moving target.
/GS/	
KNEXT(MAXSAT,MAXGP)	Flag for next position in GS
MAXGS	Flag for end of GS
GS(MAXSAT,MAXHIT,MAXGP)	Table of hit start and stop times
IGS(MAXSAT,MAXHIT/2,MAXGP)	Table of ground stations for each GS entry
/GRIDIN/	
BLAT	Beginning latitude for target grid
BLON	Beginning longitude for target grid
ELAT	Ending latitude for target grid
ELON	Ending longitude for target grid
SPALAT	Latitude spacing
SPALON	Longitude spacing
NUMLAT	Number of rows in target grid
NUMLON	Number of columns in target grid
NUMTAR	Number of targets in grid
LASTK	Counter to resume grid
LASTI	Counter to resume grid
LASTJ	Counter to resume grid
RADII	Radius of targets
/OUTPUT/	
OUTPUT(100,5)	Output buffer for OUTBUF in DELAY
NEXT(MAXGP)	Counter for OUTPUT
/ELG/	
ELG(MAXG,6)	Ground station and target elements in earth-fixed cartesian coordinates. X,Y,Z position (normalized), radius, and, for ground stations, elevation angle (station mask) in radians and a constant related to the coverage. For targets, a flag indicating the type of target.

	1 = stationary 2 = grid element 3 = ship track on great circle 4 = moving target input from file
ELGS(MAXG,4)	Ground station and target elements; latitude, longitude, and radius, in degrees For ground stations, elevation angle, in degrees; or for targets, a flag indicating type of target (see above).
HITAB(MAXSAT,MAXGP)	Table of cosines of the maximum earth central angle a satellite can be from a target and still see the point. Used for a quick comparison test.
TRG(MAXGP,5)	Input moving target elements: TRG (N,1) = beginning latitude of (moving) target N TRG (N,2) = beginning longitude of target N TRG (N,3) = ending latitude of target N TRG (N,4) = ending longitude of target N TRG (N,5) = speed of target N Positions in degrees, and speed of target in knots. If speed = 0, speed is calculated so trip takes duration of run.
/ELSAT/	
ELSAT(MAXSAT,16)	Satellite elements:
1. a[km]	Semimajor axis
2. e	Eccentricity
3. i [rad]	Inclination (input in degrees)
4. M [rad]	Mean anomaly (input in degrees)
5. ω [rad]	Argument of perigee (input in degrees)
6. Ω [rad]	Longitude of ascending node (input in degrees) Can be input as longitude or as right ascension (keyword LONG)
7. Time rate of change of 5	
8. Time rate of change of 6	
9. $\sqrt{\mu/a^3}$	
10. (Satellite epoch) – (Run epoch)	
11. Flag for whether satellite needs to see a ground station	0 = no 1 = yes
12. Antenna type code	1 = elevation angle 2 = co-dip angle 3 = earth central half-angle 4 = half swath width in nautical miles 5 = annulus 6 = side-looking antenna
13. Variable related to maximum viewing angle	
14. Antenna parameter	
15. Antenna parameter	

/MATRX/	
PSTART	Time to start first period
DELTAP	Number of time step in each period
NUMP	Number of periods
AFRAC	Minimum acceptable fraction of a period to count as a sighting for that period.
/DELTOT/	
DIFFMX	Maximum delay between two systems
DIFFMN	Minimum delay between two systems
DIFTOT	Sum of differences between two systems
NUMDIF	Total number of delays between two systems
DIFMAX(I)	Maximum delay for Ith revisit
DIFMIN(I)	Minimum delay for Ith revisit
DIFFTT(I)	Sum of delays for Ith revisit
NUMDF(I)	Number of Ith revisits between two systems

5.0 ERROR TRAPS

The program has several error-checking "traps" that are designed to recognize common errors in inputs or in the mode of running. Most "nonsense" inputs will be recognized. The program also checks for overflows in the internal storage. These traps are described below.

Condition	Result
1. Satellite epochs or run start or stop times before 1950 or after 1999	Error message printed, program stops [IDATE].
2. DT negative or zero	Error message printed, program stops [INPUT].
3. Run stop time occurs before start time.	Error message printed, program stops [INPUT].
4. Number of satellites exceeds the number the program is dimensioned for.	Error message printed, program stops [INPUT].
5. Number of targets and ground stations is greater than the number that the program is dimensioned for (using TARINPUT).	Error message printed, program stops [INPUT].
6. Number of system 1 satellites is greater than the number that the program is dimensioned for.	Error message printed, program stops [INPUT].
7. Number of revisits in DELAY is less than 0.	Error message printed, program stops [INPUT].

8.	Number of satellite hits greater than the number the program is dimensioned for.	Error message printed, program stops [COREL8].
9.	(Run time/time step) larger than maximum integer used for counting.	Warning message printed, program commences post-processing with new stop time.
10.	Keyword not recognized	Error message printed, program stops [INPUT].

6.0 REDIMENSIONING

LORI is designed to efficiently handle runs with a variety of satellites, targets, and ground stations. LORI does this by redimensioning the program to fit the particular set of runs in the computer in a manner that will ensure maximum speed of execution. Four parameters determine the size of the program. They are the number of satellites (parameter MAXSAT), the number of targets (parameter MAXGP), the number of ground stations and targets together (parameter MAXG), and twice the maximum number of sightings of each target by each satellite (parameter MAXHIT). In a core-limited machine with a program that just fits into the core, increasing one of the above parameters necessitates decreasing one or more of the others. How this can be done is explained below.

Most of LORI is taken up by two integer arrays, GS and IGS. GS consists of pairs of times, the start time step of a sighting of a target by a satellite, and the first time step afterward that the satellite does not see the target. GS is indexed by satellite, sighting number, and target, e.g., GS(3,21,4) would contain the start time (in time steps) for the eleventh sighting of target 4 by satellite 3. Time steps are measured from the epoch of the computer run, not the epoch of the satellite elements. IGS is referenced similarly and contains the number of the ground station in view at the time of that particular sighting. Due to the algorithm (it quits as soon as it finds a ground station), if two ground stations are in view, IGS will contain the index of the lowest numbered station. For the example above, the IGS element that refers to that sighting would be IGS(3,11,4).

The maximum dimensions of GS and IGS are set by the parameters MAXSAT, MAXHIT, and MAXGP. MAXSAT is the total number of satellites, system 1 plus system 2 (if present). MAXGP is the total number of targets, and MAXHIT is twice the total number of sightings for a given satellite-target pair.

Several points must be considered when choosing values for the parameters. These points involve machine size, program speed, and method of running. There are several tradeoffs the user can make. A frequent mode of running LORI involves a series of computer runs looking at some set number of targets (input explicitly and/or a grid), for some specific period (e.g., all runs are for one week with a 1-min time step), and varying the number and phasing of the input satellites up to some maximum number of satellites. In this mode, the user would set MAXGP for the maximum number of targets used, MAXSAT at the maximum number of satellites that he expects to use, and MAXHIT as large as he can and still have the program fit the machine. Having once compiled and task-built the program, the user then makes his series of runs. Problems arise when the program task does not fit the machine or during a run when the program prints an error message concerning internal storage and then exits.

LORI propagates all of the satellites at the same time, for as many targets as it can. Given a target grid of 300 points and dimensioned for 300 targets, it will check each satellite against each of the 300 targets at each time step. If dimensioned for only 10 targets, LORI will run all of the satellites for the entire program run for the first 10 targets, write out the results, and then take the next 10 targets and start over again. It will cycle by the maximum number of targets it can handle until it is through. Since most of the computer time is taken up with satellite element propagation, it is more efficient to check as many targets at the same time as is possible. For example, dimensioning for 10 targets and running a grid of 11 points will take almost twice as long as dimensioning for 11 targets and making the same run. Thus, for maximum efficiency in speed one should set the parameter MAXGP as large as the expected number of targets. If this results in the program being too large to be linked, or if an error message concerning storage is printed during execution, the user can trade off speed for storage by decreasing the number of targets per run (MAXGP), with an increase in the length of the run (MAXHIT), if necessary.

MAXHIT is twice the maximum number of sightings of any target by any satellite. For instance, for a group of satellites that have a period of 90 minutes (16 orbits per day) and a large antenna coverage, one could expect 14 to 16 sightings per day per satellite, maximum. MAXHIT for this type of satellite would then be (16*2*(number of days of run)). The user may notice cases where a version of LORI has run fine for a set of satellites, and with minor changes to the data, no longer has enough core. This is usually due to the changed configuration causing more sightings per day for the satellites (e.g., increased antenna coverage, ground targets moved, or better positioning of ground stations).

The program REDIM was written to make most of these decisions and their implementation easier for the user. REDIM prompts the user for information on the type of runs he intends to make. REDIM calculates the largest MAXHIT that will allow the program to still fit in the machine. REDIM will then create command files (these could be rewritten for other machines) that will edit the source files, using the values for the parameters it has computed, and will compile, task-build, and clean the directory of the intermediary files. LORI is then ready to run.

An example of the use of REDIM is in the sample input section.

7.0 PROGRAM CHANGES

7.1 Changes Dependent on Computer System

7.1.1 Word Size

There are two computer-dependent constants in the program. (LORI)/CONS/LARGE is the largest integer allowed for the default integer size (e.g., for the DEC 11/70 it is 32767, for the DEC VAX 11/780 it is 2147483647). This number determines the maximum number of time steps allowed. Thus with LARGE=32767 and a time step of 1 min, the longest program run time allowed is 22 days (32767/1140). On systems that allow only I*2 integers, the situation can be improved by a factor of 2 by using an offset from -32767. This would involve changes in the program wherever reference to the GS table occurs, however. The opposite consideration is that one may want to use I*2 rather than I*4 to allow space for more satellites and targets to be processed at the same time.

The other constant is (PREDIC)TOL. This is the tolerance for the interative solution to Kepler's equation. This could be tightened if running in double precision, but is probably best left at its current value (2.0×10^{-6}).

7.1.2 Compiler Differences

Converting LORI to a system different from the DEC PDP 11/70 or VAX 11/780 will probably require that the user check the following compiler or system idiosyncrasies in programming:

1. Characters per word—current value is four.

2. File OPEN and CLOSE statements—different systems may require operating system file definition statements for the files to be opened.

3. Variable type BYTE—used only for file names for the OPEN statements.

4. List-directed I/O—used only in Subroutine INPUT. Other systems may have their own equivalent, such as NAMELIST.

7.1.3 Core Size

LORI can handle large numbers of satellites, targets, and ground stations for long program runs, with the limitation of the size of program that will fit in core.

See the section on redimensioning for a discussion of adjusting the parameters to maximize the efficiency of the allowed core. The program listing and description show methods of running to effectively increase core. This section will address actual changes to the program.

Due to the structure of the program, overlaying is quite easy, with practically no penalty in execution time. In cases where a relatively small increase in core size is needed, overlaying is an obvious solution. For systems similar to the DEC PDP 11/70, another simple change to increase the core is to use a shared global common. Specifying Integer*2 for the large table GS will double the available space at the penalty of a maximum of 32767 time steps.

For extremely core-limited systems, a more drastic change would be to remove the GS table from the core completely and have it written out to a file (or files) instead. Since the table is created sequentially in time for each satellite, and the current postprocessors all process it sequentially, this is possible with a minimum penalty in time or programming. It would also obviate the need for double indexing in the postprocessors, with a simple read command rather than a doubly indexed array fetch. A suggested method is to open one file per satellite-target pair, using units 99 on down. A feature of this method is that it allows saving the tables after a run for future processing without regenerating the data.

7.2 Satellite-Dependent Changes

7.2.1 Ephemeris Generator

LORI currently uses an analytic ephemeris generator that includes the secular effects of the earth's oblateness J_2 on the ascending node and argument of perigee. The accuracy could be improved easily by implementing a more accurate analytic generator, or even a numerical integrator. Subroutine ORBIT calls PREDIC with the satellite elements as input and the position vector as output. The change would involve merely replacing Subroutine PREDIC, with possibly a change in the calling arguments. It should be noted that LORI is intended as an analytical tool and not as a case-by-case simulator of actual events. The current analytic ephemeris generator is accurate enough for

the statistical results LORI is intended to supply. The current method allows the program epoch to be before or after the epochs of the individual satellites, and the difference can be by any amount with no effect on the run time of the program. Use of a numerical integrator would greatly affect the run time of the program. Replacing PREDIC is recommended only for cases outside of its accuracy domain, such as the need to include atmospheric drag.

7.2.2 Antenna Patterns

The antenna patterns currently calculated by LORI are limited to nadir-looking full and annular cones looking at points or circles on the ground, and a side-looking, one-dimensional beam. More complicated patterns can be installed by changing Subroutine ORBIT at the area marked in the comments, or by replacing the section marked by a call to a subroutine. Currently, the routine does a quick check on the antenna footprint before doing extensive calculations. The check is merely a dot product between the satellite position vector and the target position vector. This defines the cosine of the angle between the two points (the vectors are normalized). This angle is compared to the maximum angle the satellite could have and still see the target. The latter is calculated at the start of the run. If the current angle is greater than the maximum allowable angle (in actuality the test is on dot products), there is no need to do an exact calculation. It is recommended that this technique be continued; if otherwise, the program will spend the majority of its time in this section of code.

7.2.3 Postprocessor

Probably the most common form of change a user will wish to make to LORI is to write a specialized postprocessor routine to analyze the data stored in the GS table according to some special algorithm. LORI currently will process the GS table for delays between two systems of satellites and for simultaneous coverage between two systems of satellites. The output can also be used to analyze delays between sightings for a single multisatellite system, as the sample output demonstrates. Modifying LORI for specific needs, such as simultaneous coverage between three or more systems, particular sequences of events, and so on, is quite simple and will have minimal effect on the rest of the program. Only three changes to the program are required. The new postprocessor routine needs to be added (in small computers it is a new overlay node), a new keyword is added to routine INPUT at the place marked, with an addition of a logical flag to common /TRUTH/, and the call to the routine is added to the main program LORI. The call should be inserted before the call to SIMUL, as SIMUL rewrites the GS table in a merged format for the two systems, in order to determine system overlaps. DELAY, SIMUL, and the routine can all be called in the same run as long as the new routine does not change the GS table. If the user is using REDIM to compile and link, REDIM will have to be modified similarly to account for the new routine.

As an example, a routine to show all sightings of targets by more than one satellite at the same time will be added to LORI. Note that this sample routine uses the variable type CHARACTER and will run only on machines with this type. It would not be difficult to modify it to run on other machines, however. The routine is called MULTI, will be called with a keyword named MULTI, and has as associated input one variable, MINSATS, which is the minimum number of satellites viewing the target at the same time. If a number of satellites greater than or equal to MINSATS sees the target at the same time, the routine will print out the time interval and the satellites that saw the target. The routine is listed on page 102, and a short sample output from the routine is given on page 105.

As listed above, the steps are as follows:

1. Add the keyword to call the routine into INPUT, and add a logical flag to common TRUTH to pass the information to LORI. After the line in INPUT marked

```
C*** INSERT THE NEW KEYWORD SECTION HERE
```

the user inserts the following section code:

```
320     IF (WORD.NE. 'MULTI') GO TO 325
        MULTY=.TRUE.
        READ(7,*)MINSATS
        GO TO 50
```

and changes the next line to

```
325     IF (WORD.EQ. 'START') GO TO 330
```

At the beginning of each routine the user adds the line LOGICAL MULTY and adds MULTY to the common TRUTH.

2. The section in LORI marked

```
180     CONTINUE
190     IF(.NOT.SIMULT)GO TO  210
```

is changed to

```
180     IF(.NOT.MULTY)GO TO 190
        DO 185 I=1,NGP
        NUMFIL=7+I
        IF(SINGL)NUMFIL=8
185     CALL MULTI
190     IF(.NOT.SIMULT)GO TO 210
```

and MULTY is data initialized to FALSE.

3. MINSATS can be added to an existing common, or a new common can be added for input data for MULTI. The common should be in INPUT and MULTI (in the listing it is: COMMON/MULTIP/MINSATS).

4. If changing REDIM, the user need only change the commented lines in REDIM and add the name of the new postprocessor in the DATA statement. The LINK statement should have the new processor name added, of course.

The use of MULTI in the input deck would be to add the following to the keyword section (the user wants all times listed):

```
MULTI
0
```

Note that in the example the user wanted only those times listed when one or more satellites were in view.

8.0 PROGRAM LISTING

8.1 Program LORI

```
C***
C***  PROGRAM LORI(LOCATOR OF REPEATING INCIDENTS) IS A
C***  GENERAL PURPOSE TIME-COVERAGE PROGRAM.
C***  IT WILL CALCULATE TIME DELAYS BETWEEN SIGHTINGS AND/OR
C***  SIMULTANEOUS COVERAGE FOR ONE OR TWO SYSTEMS OF SATELLITES.
C***  ANY SATELLITE IN EITHER SYSTEM MAY REQUIRE SIMULTANEOUS SIGHTING
C***  OF A GROUND STATION AND A TARGET TO REGISTER A SIGHTING.
C***  THE SIZE OF SYSTEMS, TIME OF RUN, AND NUMBER OF TARGETS THE
C***  PROGRAM WILL HANDLE ARE DETERMINED BY THE PARAMETERS:
C***  MAXGP=MAXIMUM # TARGETS,  MAXG=MAXIMUM # TARGETS AND STATIONS
C***  MAXSAT = MAX # OF SATELLITES IN BOTH SYSTEMS,
C***  MAXHIT = TWICE THE MAXIMUM # OF HITS OF ANY SATELLITE TO ANY
C***  TARGET.
C***  USING THE @A MODE THESE PARAMETERS CAN BE CHANGED AUTOMATICALLY.
C***  THIS IS DONE BY TYPING @A IN THE MONITOR.
C***  FOR THE DELAY MODE, A MINIMUM AND MAXIMUM TIME DELAY ARE INPUT,
C***  AS IS THE NUMBER OF REVISITS OF SYS2 TO A GIVEN SYS1 HIT.
C***  NOTE THAT DELAYS ARE SYS1 TO SYS2(SYS1 FIRST).
C***  THE SATELLITES NEED NOT BE RELATED IN ANY WAY IN REGARDS TO
C***  ELEMENTS, WHETHER BETWEEN SYSTEMS OR INTRA-SYSTEM.
C***  INPUT IS FREE FORMAT, AND IN A PARTICULAR ORDER, VIZ. :
C***
C***
C***  2.           TIME STEP SIZE IN MINUTES(OR SECONDS WITH KEYWORD)
C***  78 6 5 0 0 0  START TIME EPOCH-YY MM DD HH MM SEC.
C***  78 6 10 0 0 0 STOP  TIME EPOCH-DITTO
C***  2              NUMBER OF SATELLITES IN SYS1(MAXIMUM MAXSAT)
C***  2              NUMBER OF SATELLITES IN SYS2(MAXIMUM MAXSAT-NSAT1)
C***  2              NUMBER OF GROUND STATIONS(MAXIMUM MAXG-MAXGP)
C***  30 45  0       LATITUDE, LONGITUDE, AND RADIUS OF STATIONS
C***  45. 45. 0.     SATELLITE ELEMENTS FOLLOW(SEE BELOW)
C***  6933 .0  63. 240. 120. 0. 78 6 5 0 0 0  0. 1.
C***  40.
C***  6933 .0  63. 240. 300. 0. 78 6 5 0 0 0  0. 1.
C***  40.
C***  7450 .001 63. 120. 300. 0. 78 6 10 0 0 0 0. 1.
C***  5.
C***  7450 .001 63. 120. 120. 0. 78 6 10 0 0 0 0. 1.
C***  5.
C***  DELAY          KEYWORD FOR DELAY CALCULATION
C***  2              NUMBER OF HITS OF SYS2 TO SYS1(NO LIMIT ON MAXIMUM)
C***  60.            MINIMUM DELAY TIME IN MINUTES
C***  420.           MAXIMUM DELAY TIME IN MINUTES
C***  SINGLFIL       PUT ALL OUTPUT ON ONE FILE WITH ONE MAIN HEADER
C***  DEBUGOUT       PRINT OUT INTERNAL WORK TABLE
C***  SIMUL          CALCULATE SIMULTANEOUS COVERAGE
C***  SHORTOUT       SIMULTANEOUS COVERAGE IS WITH SHORT OUTPUT
C***  SYSTMCOV       PRINT OUT COVERAGE TIMES FOR THE TWO SYSTEMS(SIMUL)
C***  TOTAL          FOR SIMUL, MERGE BOTH SYSTEMS INTO ONE
C***  MATRIX         RUN MATRIX POST-PROCESSOR
```

```
C***  1 3Ø 6Ø 1.Ø    PERIOD START,PERIOD LENGTH,NO. PERIODS, ACCEP. FRAC
C***  TARINPUT       TARGET ELEMENTS ARE INPUT AND FOLLOW(LAT,LONG,RAD)
C***  2              NUMBER OF TARGETS INPUT
C***  4Ø. 5Ø. Ø.
C***  2Ø.  Ø. Ø.
C***  TARGRID        TARGET ELEMENTS ARE A GRID AND THE ELEMENTS FOLLOW
C***  Ø. Ø. 9Ø. Ø. 1Ø. Ø. 5. (FROM Ø,Ø TO 9Ø,Ø BY 1Ø,Ø RADIUS=5 DEG.)
C***  SHPTRK         TARGET ELEMENTS ARE MOVING ON GREAT CIRCLES
C***  1              NUMBER OF SHIPTRAK TARGETS
C***  Ø. Ø. 2Ø. 3Ø. 15.  START LAT,LON, END LAT,LON, SPEED IN KNOTS
C***  MOVETARG       TARGETS ARE MOVING , TRACKS ARE READ FROM FILES
C***  2              NUMBER OF MOVETARG TARGETS
C***  FILENAME#1     FILE NAME OF FIRST MOVING TARGET
C***  FILENAME#2     FILE NAME OF SECOND MOVING TARGET
C***  NUMHIT        PRINT OUT # OF HITS FOR EACH SAT ON TARGET
C***  VARIATIO       USED FOR A SATELLITE GRID.  WILL TAKE SAT # 1
C***  3 Ø. 1Ø. 1Ø    AND VARY ELEMENT # 3 FROM Ø. TO 1ØØ. BY 1Ø, EG.
C***  SECONDS             TIMES ARE INPUT IN SECONDS
C***  MEAN                SATELLITE INPUT USES MEAN ANOMALY(DEFAULT)
C***  TRUE                SATELLITE INPUT USES TRUE ANOMALY
C***  START
C***
C***
C***
C***  THE THIRD ELEMENT OF THE TARGETS AND GROUND STATIONS IS THE
C***  RADIUS, IN DEGREES, OF THE TARGET OR GROUND STATION. IF THIS
C***  IS POSITIVE, IT DEFINES A CIRCLE CENTERED ON THE TARGET OR
C***  GROUND STATION, ALL OF WHICH MUST BE SEEN BY THE SATELLITE
C***  TO REGISTER AS A HIT. IF IT IS NEGATIVE, IT DEFINES A CIRCLE OF
C***  WHICH ANY CAN BE SEEN BY A TARGET TO REGISTER AS A HIT. IF
C***  IT IS ZERO IT DEFINES A POINT.
C***  THE AREA COVERED INCREASES WITH ALTITUDE OF THE SATELLITE
C***  FOR ELEVATION ANGLE OR CONE ANGLE INPUT, BUT DOES NOT CHANGE
C***  FOR EARTH CENTRAL ANGLE OR SWATH WIDTH.  THE INCREASE IN COVERAGE
C***  WITH INCREASE IN ALTITUDE IS NOT THE SAME FOR ELEVATION OR CONE
C***  ANGLE INPUT.  BE SURE YOU KNOW WHAT YOU INTEND!
C***  NOTE THAT THE INPUT IS FOR HALF-ANGLES FOR ALL BUT ELEVATION,
C***  EG.,HALF CONE-ANGLE, HALF SWATH-WIDTH, AND HALF EARTH ANGLE.
C***  THE SATELLITE ELEMENTS ARE IN DEGREES AND KILOMETERS AND ARE:
C***  A(KM),E,I(DEG),MEAN(OR TRUE) ANOMALY(DEG), ARG OF PERI(DEG),
C***  AND R.A.(OR LONG)OF THE ASC. NODE(DEG).  THESE ARE FOLLOWED BY
C***  THE EPOCH OF THE ELEMENTS, GIVEN AS YY MM DD HH MM SEC.,
C***  AND THE THE FLAG FOR WHETHER THE SATELLITE NEEDS TO SEE A
C***  GROUND STATION(Ø=NO,1=YES), AND THE ANTENNA TYPE.
C***  THE ANTENNA TYPES ARE:1=ELEVATION ANGLE, 2=CO-DIP ANGLE,
C***  3= EARTH CENTRAL ANGLE, 4= SWATH HALF-WIDTH IN NM.,
C***  5= ANNULAR RING, 6= SIDE-LOOKING BEAM.
C***  THE PARAMETER(S) FOR THE ANTENNA FOLLOW ON THE NEXT LINE,
C***  AND ARE IN DEGREES OR NM.  FOR THE ANNULAR RING THE FIRST
C***  PARAMETER IS THE CO-DIP OF  THE OUTER LIMIT, THE SECOND
C***  IS THE CO-DIP OF THE INNER LIMIT.  FOR THE SIDE BEAM THE
```

```
C***  FIRST PARAMETER IS THE CO-DIP OF THE BEAM BORESIGHT, THE
C***  SECOND PARAMETER IS THE BEAM WIDTH(+CODIP=PORT,-CODIP=STARBOARD).
C***  THE OUTPUT IS ON A SEPARATE FILE FOR EACH TARGET. IT CONSISTS
C***  OF A HEADER WHICH REPRODUCES THE INPUT, AND THEN THE ASSOCIATED
C***  OUTPUT.  THE OUTPUT IS ON FOR008.DAT,FOR009.DAT,ETC
C***  FOR TARGET ONE, TWO, ETC.
C***  IF SINGLFIL IS USED, ALL OUTPUT WILL BE ON FOR008.DAT, WITH
C***  ONLY ONE HEADER.
C***  TARGRID WILL HAVE OUTPUT JUST LIKE A REGULAR RUN, WITH OR
C***  WITHOUT SINGLFIL.  HOWEVER, IF THE NUMBER OF ELEMENTS IN THE
C***  GRID IS LARGER THAN THE MAXIMUM
C***  DIMENSION FOR THE TARGETS(MAXGP), THE OUTPUT WILL
C***  BE ON FILES FROM FOR008.DAT TO FOR0(8+MAXGP).DAT, AND THEN
C***  REPEAT ON FOR008.DAT TO *, WITH A NEWER EXTENSION.THE PROGRAM
C***  WILL DO AS MANY TARGETS AT ONE TIME THAT IT CAN.  THIS CAN BE
C***  OPTIMIZED USING THE @A MODE.
C***
C***
C***  THE HEADER IS REPEATED, THE SECOND HEADER MISSING THE DATA
C***  ON THE GROUND STATIONS AND THE SATELLITES. IF THIS DATA IS
C***  SENSITIVE, THE FIRST HEADER CAN BE RIPPED OFF, LEAVING ENOUGH
C***  INFORMATION WITH THE OUTPUT TO DISTINGUISH IT.
C***
C***  THE KEYWORD ENTRIES CAN BE IN ANY ORDER AFTER TMAX WITH THE
C***  EXCEPTION OF START, WHICH MUST BE THE LAST.
C***  TMIN MAY BE ZERO OR EVEN NEGATIVE, IF THATS WHAT YOU WANT.
C***  TMAX CAN BE ANY VALUE- IT DOES NOT AFFECT RUN TIME.
C***  IF USING VARIATIO ONLY SAT # 1 WILL BE PRINTED OUT
C***  (AND ONLY 1 SAT SHOULD BE INPUT, AND NSAT1 SHOULD BE 1)
C***  THIS IS TO CUT DOWN ON THE AMOUNT OF PRINTOUT AS ONE COULD
C***  BE USING 360 SATELLITES, FOR EXAMPLE, AND ONLY ONE ELEMENT
C***  IS CHANGING.
C***
C***  MATRIX RUNS THE MATRIX POST-PROCESSOR. THIS DIGITIZES THE COVERAGE
C***  INTO USER-DEFINED PERIODS.  THE USER INPUTS THE TIME STEP THE
C***  FIRST PERIOD STARTS FROM, THE NUMBER OF TIME STEPS IN A PERIOD,
C***  THE NUMBER OF PERIODS TO USE, AND THE ACCEPTABLE FRACTION OF THE
C***  PERIOD A SATELLITE CAN SEE THE TARGET AND STILL COUNT.
C***  THE OUTPUT FROM MATRIX IS A MATRIX, WITH THE PERIOD AS THE ROW
C***  AND THE SATELLITE AS THE COLUMN.  IT PRINTS OUT THE MATRIX WITH
C***  A BLANK FOR THE SATELLITE-PERIOD PAIR IF THE SATELLITE SAW THE
C***  TARGET FOR LESS THAN THE ACCEPTABLE FRACTION OF THE PERIOD, AND
C***  PRINTS A 1 IF THE SATELLITE SAW THE TARGET FOR MORE THAN THE
C***  ACCEPTABLE FRACTION.  THE ROUTINE ALSO PRINTS OUT THE ROW AND
C***  COLUMN TOTALS, IE., THE TOTAL NUMBER OF SATELLITES THAT SAW THE
C***  TARGET FOR THAT TIME PERIOD, AND THE TOTAL NUMBER OF PERIODS
C***  EACH SATELLITE SAW THE TARGET.
C***  MATRIX HAS SEVERAL USES, BUT ONE IS AS A PRINTER GRAPHICS OUTPUT
C***  OF THE COVERAGE.
C***
C***
```

```
C***
C***
C***
C***
C***
C***
C***
C***
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
C***
      DIMENSION KN(MAXSAT,MAXGP)
C***
      DATA OUTPUT,NEXT,LARGE/600*0,MAXGP*1,2147483647/
      DATA SIMULT,SINGL,SHORT,GRID,DBUG,SYSTM,NUMHIT/7*.FALSE./
      DATA MAT,ALLDUN,IFDONE/2*.FALSE.,MAXGP*.FALSE./
      DATA PI/3.14159265358979324D0/
      DATA XMU/398603.2D0/,XJ2/.001082645D0/, RE/6378.165D0/
      DATA WE/.729212D-4/,NREP/1/,INP/0/
      DATA INC/1/
C***
C***  IGS IS DIMENSIONED 1/2 OF GS IN THE MIDDLE DIMENSION
C***  MAXGS IS SET TO THE MAXIMUM DIMENSION OF GS-THIS IS TO LET THE
C***  PROGRAM KNOW WHEN IT IS ABOUT TO OVERWRITE CORE.
C***  IF CHANGING DIMENSIONS OF GS AND IGS, BE SURE TO CHANGE THE
C***  DIMENSIONS IN ALL APPLICABLE SUBROUTINES, THE SIZE OF THE DATA
C***  STATEMENTS, AND THE VALUE FOR MAXGS.
```

```
C***  NOTE THAT SIMUL USES EQUIVALENCES FOR IGS AND GS.
C***  IF ADDING A POST-PROCESSOR DO IT BEFORE SIMUL.
C***
C***    TIMRB AND TIMRE ARE SYSTEM TIMING ROUTINES
C***
      CALL TIMRB
      TPI=2.DØ*PI
      RTD=18Ø.DØ/PI
      DTR=PI/18Ø.DØ
      MAXGS=MAXHIT-2
C***
C***INPUT READS IN DATA
C***
      CALL INPUT
C***
C***   SETUP CREATES TABLES THAT ARE CALCULATED ONLY ONCE
C***
      CALL SETUP
   1Ø TIME=Ø.
C***
C*** START LOOP TO CALCULATE WORK TABLE OF SAT. START AND STOP TIMES
C*** OF DETECTIONS
C***
   2Ø CALL COREL8
      IF(ALLDUN) GO TO 4Ø
      TIME=TIME+DT
   3Ø IF(TIME.LE.TSTOP) GO TO 2Ø
   4Ø CONTINUE
C***
C***  THE FOLLOWING LOOP IS FOR SHIP-TRACK TARGETS THAT REACH
C***  THEIR DESTINATION BEFORE THE PROGRAM RUN ENDS.
C***
      DO 5Ø I=1,NGP
   5Ø IF(ISTOP(I).EQ.LARGE)ISTOP(I)=TSTOP/DT
   6Ø CONTINUE
      DO 7Ø I=1,NGP
      IF(ISTOP(I).EQ.INT(TSTOP/DT)) GO TO 7Ø
      LOOP=NGP+7
      IF(SINGL)LOOP=8
      OPEN(UNIT=LOOP,TYPE='OLD',ACCESS='APPEND')
      WRITE(LOOP,39Ø)I,ISTOP(I)
      TIMWRT=ISTOP(I)*DT
      WRITE(LOOP,4ØØ)TIMWRT
      CLOSE(UNIT=LOOP)
   7Ø CONTINUE
   8Ø CONTINUE
C***
C***   THE FOLLOWING DO LOOP FINISHES OFF THE INTERNAL WORK FILE
C***   BY THAT I MEAN IF A SATELLITE IS SEEING A TARGET AT TSTOP
C***   IT PUTS TSTOP INTO GS AS THE FINAL PART OF A START-STOP PAIR.
C***   IT ALSO INSURES THAT THE NEXT ENTRIES IN GS ARE LARGE.
```

```
C***   THIS IS IMPORTANT FOR MULTI-PASS RUNS(GRIDS).
C***
      DO 90 J=1,NGP
      DO 90 I=1,NTOT
      IF(GS(I,KNEXT(I,J),J).EQ.LARGE) GO TO 90
      GS(I,KNEXT(I,J)+1,J)=ISTOP(J)+1
      KNEXT(I,J)=KNEXT(I,J)+2
      GS(I,KNEXT(I,J),J)=LARGE
      GS(I,KNEXT(I,J)+1,J)=LARGE
   90 CONTINUE
      IF(MAT) CALL MATRIX
      IF(.NOT.DELA) GO TO 120
C***
C***  THIS PROCESSES THE GS FILE AND CALCULATES DELAYS
C***  BETWEEN SYSTEM ONE AND SYSTEM TWO
C***  OUTPT FLUSHES OUTPUT BUFFER
C***
      DO 110 NUMGP=1,NGP
      NFILE=NUMGP+7
      IF(SINGL)NFILE=8
      OPEN(UNIT=NFILE,ACCESS='APPEND',TYPE='OLD')
      CALL DELAY
      CALL OUTBUF
      NUMDIF=MAX(1,NUMDIF)
      IF(DIFFMN.GE.1.E30)DIFFMN=0.
      DIFAVG=DIFTOT/NUMDIF
      WRITE(NFILE,380)DIFFMX,DIFFMN,DIFAVG,NUMDIF
      DO 100 I=1,INC
      NUMDF(I)=MAX(1,NUMDF(I))
      IF(DIFMIN(I).GE.1.E30)DIFMIN(I)=0.
      DIFAVG=DIFTT(I)/NUMDF(I)
      WRITE(NFILE,370)I,DIFMAX(I),DIFMIN(I),DIFAVG,NUMDF(I)
  100 CONTINUE
      CLOSE(UNIT=NFILE)
  110 CONTINUE
  120 IF(.NOT.NUMHIT) GO TO 150
C***
C***   THIS PRINTS OUT THE NUMBER OF HITS FOR EACH SAT
C***   KNEXT IS ACTUALLY ((#HITS+1)*2+1)
C***
      DO 140 NUMFIL=1,NGP
      NAMFIL=NUMFIL+7
      IF(SINGL) NAMFIL=8
      OPEN(UNIT=NAMFIL,TYPE='OLD',ACCESS='APPEND')
      IF(SINGL.AND.NUMFIL.EQ.1) WRITE(NAMFIL,330)
      DO 130 IKY=1,NTOT
  130 KN(IKY,NUMFIL)=(KNEXT(IKY,NUMFIL)-1)/2
      WRITE(NAMFIL,360)NUMFIL,(KN(I,NUMFIL),I=1,NTOT)
  140 CLOSE(UNIT=NAMFIL)
  150 IF(.NOT.DBUG) GO TO 190
C***
```

```
C***   THIS PRINTS OUT THE INTERNAL TABLE FOR DEBUG PURPOSES
C***
      DO 18Ø IK=1,NGP
      IMAX=Ø
      NFILE=IK+7
      IF(SINGL)NFILE=8
      OPEN(UNIT=NFILE,ACCESS='APPEND',TYPE='OLD')
      DO 16Ø IKI=1,NTOT
  16Ø IMAX=MAX(IMAX,KNEXT(IKI,IK))
      IF(.NOT.SINGL)WRITE(NFILE,28Ø) IK
      IF(SINGL.AND.IK.EQ.1)WRITE(NFILE,35Ø)
      IF(SINGL)WRITE(NFILE,34Ø) IK
      WRITE(NFILE,3ØØ)(IKTOT,IKTOT=1,NTOT)
      DO 17Ø J=1,IMAX,2
      IJ=J/2+1
      WRITE(NFILE,29Ø)  (GS(K,J,IK),K=1,NTOT)
      WRITE(NFILE,29Ø)  (GS(K,J+1,IK),K=1,NTOT)
      WRITE(NFILE,29Ø)  (IGS(K,IJ,IK),K=1,NTOT)
  17Ø CONTINUE
  18Ø CLOSE(UNIT=NFILE)
C***
C***     IF ADDING A NEW POST-PROCESSOR, THE CALL SHOULD
C***     GO HERE, BEFORE SUBROUTINE SIMUL IS CALLED.
C***     SIMUL MODIFIES THE GS TABLE FROM ITS ORIGINAL FORM.
C***
  19Ø CONTINUE
  2ØØ IF(.NOT.SIMULT) GO TO 22Ø
C***
C***  THIS SECTION PROCESSES THE INTERNAL TABLE TO FIND SIMULTANEOUS
C***  COVERAGE.  NOTE THAT AFTER THIS SECTION(IF IT IS REACHED), THE
C***  INTERNAL TABLE AREA NOW CONTAINS NOT START-STOP TIMES FOR
C***  THE INDIVIDUAL SATELLITES BUT FOR THE TWO SYSTEMS.
C***  SYS1 IS MERGED INTO IGS AND SYS2 IS MERGED INTO A SEPARATE ARRAY.
C***  THEY ARE THEN BOTH MERGED INTO GS, SO BOTH GS AND IGS NO LONGER
C***  HAVE THE INDIVIDUAL START-STOP TIMES AFTER THE CALL TO SIMUL.
C***
      DO 21Ø NUMGP=1,NGP
      CALL SIMUL
  21Ø CONTINUE
  22Ø CONTINUE
C***
C***  IF THE TARGETS ARE A GRID THIS SECTION RESTARTS WITH NEW TARGETS
C***  UNTIL ALL POINTS IN THE GRID ARE COVERED.
C***
      IF(.NOT.GRID) GO TO 27Ø
      NREP=NREP+1
      IF(LASTK.GE.NUMTAR) GO TO 27Ø
      DO 23Ø I=1,MAXSAT
      DO 23Ø J=1,MAXGP
      GS(I,1,J)=LARGE
  23Ø GS(I,2,J)=LARGE
```

```
      DO 240 K=LASTK+1,NUMTAR
      I=(K-1)/NUMLON+1
      J=MOD(K-1,NUMLON)+1
      K3=MOD(K,NREP*MAXGP)
      IF(K3.EQ.1) GO TO 250
      LASTK2=K
      K2=MOD(K,MAXGP)
      IF(K2.EQ.0)K2=MAXGP
      L=K2+NGS
      ELGS(L,1)=(I-1)*SPALAT+BLAT
      ELGS(L,2)=(J-1)*SPALON+BLON
      ELGS(L,3)=RADII
  240 CONTINUE
  250 CONTINUE
      DO 260 IN=1,MAXGP
  260 LSTART(IN)=0
      LASTK=LASTK2
      NGP=K2
      NGTOT=NGS+NGP
      INP=1
      IF(.NOT.SINGL)CALL INPUT
      CALL SETUP
      GO TO 10
  270 CONTINUE
  280 FORMAT(1H1,' DEBUG OUTPUT FOR TARGET # ',I2,
     1 /,'  START OF SIGHTING, START OF NON-SIGHTING, AND STATION #')
  290 FORMAT( 12(1X,I10))
  300 FORMAT('    ',<NTOT>('  SAT # ',I2))
  310 FORMAT(1X,12(I10))
  320 FORMAT(1H0,' TOTAL COVERAGE OF TARGET BY SYS1 AND SYS2 BOTH IS',
     1 F9.1,' MINUTES')
  330 FORMAT(1H1)
  340 FORMAT(1H0,' DEBUG OUTPUT FOR TARGET # ',I2,
     1 /,'  START OF SIGHTING, START OF NON-SIGHTING, AND STATION #')
  350 FORMAT(1H1)
  360 FORMAT(' NUMBER OF HITS FOR TARGET ',I6,' FOR EACH SATELLITE =',/,
     1 12(I6,4X))
  370 FORMAT(' FOR THE ',I3,'  DELAY BETWEEN SYSTEMS ',/,
     1 ' MAXIMUM DELAY = ',F10.2,/,
     2 ' MINIMUM DELAY = ',F10.2,/,
     3 ' AVERAGE DELAY = ',F10.2,/,
     4 ' TOTAL NUMBER OF DELAYS = ',I8)
  380 FORMAT(' FOR ALL DELAYS BETWEEN SYSTEMS ',/,
     1 ' MAXIMUM DELAY = ',F10.2,/,
     2 ' MINIMUM DELAY = ',F10.2,/,
     3 ' AVERAGE DELAY = ',F10.2,/,
     4 ' TOTAL NUMBER OF DELAYS = ',I8)
  390 FORMAT(1X,'TARGET # ',I4,' STOPPED AT TIME STEP ',I12)
  400 FORMAT(1X,' AT TIME = ',F12.2,' SEC ')
      CALL TIMRE
      END
```

8.2 Subroutine INPUT

```
C***
C***  INPUT READS IN THE DATA.
C***  DUE TO SPACE CONSIDERATIONS, IT OPENS THE INPUT FILE, READS ALL
C***  DATA, AND CLOSES THE INPUT FILE BEFORE DOING ANY WRITING. THIS
C***  ALLOWS LINKING WITH ACTFIL=2, IE., ONLY TWO ACTIVE FILES AND THE
C***  ASSOCIATED BUFFERS.  SIMILARLY, ALL WRITING IS DONE TO ONLY ONE
C***  OUTPUT FILE AT A TIME.  THE OTHER OPEN FILE IS THE TERMINAL.
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      BYTE INFILE(40),TARFIL(40)
      DOUBLE PRECISION WORD,TYP(4)
      LOGICAL SEC,IFGS
      REAL TYPA(6),RA,LON,MEANL,TRUEL
C***
C***
      DIMENSION IELSAT(MAXSAT,5),SECSAT(MAXSAT)
C***
      DATA LONG,SEC/.FALSE.,.FALSE./
      DATA RA,LON /'R.A.','LONG'/
      DATA MEANL,TRUEL/'MEAN','TRUE'/
      DATA TOTAL/.FALSE./
      DATA MEAN/.TRUE./
      DATA TYP/'UNMOVING','GRID    ','SHIPTRAK','MOVING  '/
      DATA TYPA/'ELEV','CODP','CENT','DIST','ANUL','SIDE'/
      DATA ZERO/0./
C***
```

```
C***   IF NOT THE FIRST TIME THROUGH(IE.,A GRID), GO STRAIGHT TO OUTPUT
      IF(INP.GE.1) GO TO 420
      DO 10 I=1,MAXSAT
      DO 10 J=1,16
   10 ELSAT(I,J)=0.E0
      DO 20 I=1,MAXG
      DO 20 J=1,6
   20 ELG(I,J)=0.E0
C***
C*** GET INPUT FILE NAME
C***
      TYPE 770
      ACCEPT 780,INFILE
      INFILE(40)=0
      OPEN(UNIT=7,TYPE='OLD',NAME=INFILE)
C***
C*** GET TITLE
C***
      TYPE 790
      ACCEPT 800
C***
C***      FIRST OPEN INPUT FILE, READ ALL DATA, THEN CLOSE THE FILE
C***      SEE MAIN PROGRAM FOR DESCRIPTION OF INPUT
C***
      READ(7,*) DTT
      READ(7,*) ISY,ISM,ISD,ISH,ISMIN,SSEC
      READ(7,*) ISTPY,ISTPM,ISTPD,ISTPH,ISTPMI,STPSEC
      READ(7,*) NSAT1
      READ(7,*) NSAT2
      READ(7,*) NGS
      N1P1=NSAT1+1
      NTOT=NSAT1+NSAT2
      NGTOT=NGP+NGS
      NGSP1=NGS+1
C***
C*** ELG CONTAINS THE GROUND STATION AND TARGET ELEMENTS.
C*** THE GROUND STATION ELEMENTS START AT 1.
C*** THE TARGET ELEMENTS START IMMEDIATELY FOLLOWING THE GROUND
C*** STATION ELEMENTS.
C***
      IF(NGS.EQ.0)GO TO 40
      DO 30 I=1,NGS
C***
C***   READ LAT, LONG, RADIUS, ELEVATION ANGLE(STATION MASK)
C***
   30 READ(7,*) ELGS(I,1),ELGS(I,2),ELGS(I,3),ELGS(I,4)
   40 CONTINUE
      DO 70 I=1,NTOT
      READ(7,*) (ELSAT(I,K),K=1,6),(IELSAT(I,J),J=1,5),SECSAT(I),
     1 ELSAT(I,11),ELSAT(I,12)
      GO TO (50,50,50,50,60,60)ELSAT(I,12)
```

```
   50 READ(7,*)ELSAT(I,14)
      GO TO 70
   60 READ(7,*)ELSAT(I,14),ELSAT(I,15)
      GO TO 70
   70 CONTINUE
   80 READ(7,830)WORD
C***
C***   THIS IS THE KEYWORD ENTRY SECTION
C***
      IF(WORD.NE.'SIMUL') GO TO 90
      SIMULT=.TRUE.
      GO TO 80
C***
   90 IF(WORD.NE.'LONG') GO TO 100
      LONG=.TRUE.
      GO TO 80
C***
  100 IF(WORD.NE.'SINGLFIL') GO TO 110
      SINGL=.TRUE.
      GO TO 80
C***
  110 IF(WORD.NE.'SECONDS')GO TO 120
      SEC=.TRUE.
      GO TO 80
C***
  120 IF(WORD.NE.'SHORTOUT') GO TO 130
      SHORT=.TRUE.
      GO TO 80
C***
C***
  130 IF(WORD.NE.'TARGRID') GO TO 140
      GRID=.TRUE.
      READ(7,*) BLAT,BLON,ELAT,ELON,SPALAT,SPALON,RADII
      GO TO 80
C***
  140 IF(WORD.NE.'DEBUGOUT') GO TO 150
      DBUG=.TRUE.
      GO TO 80
C***
C***
  150 IF(WORD.NE.'TARINPUT') GO TO 170
      READ(7,*)NUMINP
      NGPP1=NGP+1
      NGTOT1=NGTOT+1
      NGTOT=NGTOT+NUMINP
      NGP=NGP+NUMINP
      DO 160 I=NGTOT1,NGTOT
      READ(7,*) ELGS(I,1),ELGS(I,2),ELGS(I,3)
  160 ELGS(I,4)=1.05
      GO TO 80
C***
```

```
  170 IF(WORD.NE.'SYSTMCOV') GO TO 180
      SYSTM=.TRUE.
      GO TO 80
C***
  180 IF(WORD.NE.'DELAY') GO TO 190
      READ(7,*) INC
      READ(7,*) TMINN
      READ(7,*) TMAXX
      DELA=.TRUE.
      GO TO 80
C***
C***
  190 IF(WORD.NE.'TOTAL') GO TO 200
      TOTAL=.TRUE.
      GO TO 80
C***
  200 IF(WORD.NE.'MOVETARG') GO TO 220
      READ(7,*)NUMMOV
      NGTOT1=NGTOT+1
      NGTOT=NGTOT+NUMMOV
      NGPP1=NGP+1
      NGP=NGP+NUMMOV
      DO 210 MOVT=NGPP1,NGTOT
      READ(7,780)TARFIL
      TARFIL(40)=0
      MOVFIL=100-MOVT
      OPEN(UNIT=MOVFIL,TYPE='OLD',NAME=TARFIL,FORM='UNFORMATTED')
      READ(MOVFIL)TRG(MOVT,1),TRG(MOVT,2),TRG(MOVT,3),TRG(MOVT,4)
      MOVCNT=NGS+MOVT
      ELGS(MOVCNT,1)=TRG(MOVT,1)
      ELGS(MOVCNT,2)=TRG(MOVT,2)
      ELGS(MOVCNT,3)=0.
      ELGS(MOVCNT,4)=4.05
      READ(MOVFIL)TOTIME,DELT
  210 CONTINUE
      GO TO 80
C***
  220 IF(WORD.NE.'SHPTRK') GO TO 240
      READ(7,*)NUMSHP
      NGTOT1=NGTOT+1
      NGPP1=NGP+1
      NGP=NGP+NUMSHP
      DO 230 I=NGPP1,NGP
      NIM=NGS+I
      READ(7,*)TRG(I,1),TRG(I,2),TRG(I,3),TRG(I,4),TRG(I,5)
      ELGS(NIM,1)=TRG(I,1)
      ELGS(NIM,2)=TRG(I,2)
      ELGS(NIM,4)=3.05
  230 CONTINUE
      NGTOT=NGS+NGP
      GO TO 80
```

```
C***
  240 IF(WORD.NE.'VARIATIO') GO TO 280
      READ(7,*)ITEM,START,AMOUNT,NUMVAR
      TEMPST=START
      NSAT1=NUMVAR
      NTOT=NUMVAR
      N1P1=NSAT1+1
      DO 270 IVAR=1,NUMVAR
      DO 250 IKOUN=1,16
  250 ELSAT(IVAR,IKOUN)=ELSAT(1,IKOUN)
      DO 260 IKOUN=1,5
  260 IELSAT(IVAR,IKOUN)=IELSAT(1,IKOUN)
      SECSAT(IVAR)=SECSAT(1)
      ELSAT(IVAR,ITEM)=TEMPST
      TEMPST=TEMPST+AMOUNT
  270 CONTINUE
      VARIA=.TRUE.
      GO TO 80
C***
  280 IF(WORD.NE.'NUMHIT')GO TO 290
      NUMHIT=.TRUE.
      GO TO 80
C***
  290 IF(WORD.NE.'MATRIX') GO TO 300
      MAT=.TRUE.
      READ(7,*)PSTART,DELTAP,NUMP,FRAC
      AFRAC=FRAC*DELTAP
      GO TO 80
C***
  300 IF(WORD.NE.'TRUE') GO TO 310
      MEAN=.FALSE.
      GO TO 80
C***
  310 IF(WORD.NE.'MEAN') GO TO 320
      MEAN=.TRUE.
      GO TO 80
C***
C*** IF ADDING A NEW OPTION, OR A NEW POST-PROCESSOR,
C*** INSERT THE NEW KEYWORD SECTION HERE
  320 CONTINUE
C***
  330 IF(WORD.EQ.'START') GO TO 340
      TYPE 840,WORD
      STOP
  340 CONTINUE
      CLOSE(UNIT=7)
C***
C*** END OF INPUT SECTION
C***
      TMINN=TMINN-AMOD(TMINN,DTT)
      TMAXX=TMAXX-AMOD(TMAXX,DTT)
```

```
      ISTART=IDATE(ISY,ISM,ISD)
      STSEC=ISH*3600.E0+ISMIN*60.E0+SSEC
      ITSTOP=IDATE(ISTPY,ISTPM,ISTPD)
      TSTOP=(ITSTOP-ISTART)*86400.E0+(ISTPH-ISH)*3600.E0+
     1 (ISTPMI-ISMIN)*60.E0+(STPSEC-SSEC)
C***
C***   START OF INPUT CHECK SECTION
C***   IF NONSENSE INPUT IS FOUND, PROGRAM WILL IDENTIFY THE
C***   CULPRIT AND STOP
C***
      IF(DTT.GT.0.) GO TO 350
      TYPE *,' DT MUST BE POSITIVE, AND ',DT,' IS NOT!'
      STOP
  350 CONTINUE
      IF(TSTOP.GT.0)GO TO 360
      TYPE *,' TSTOP IS NEGATIVE!'
      STOP
  360 CONTINUE
      IF(NTOT.LE.MAXSAT)GO TO 370
      TYPE *,' NUMBER OF SATELLITES GREATER THAN PROGRAM ALLOWS'
      ITEMP=MAXSAT
      TYPE *,' MAXIMUM PROGRAM ALLOWS AT PRESENT IS ',ITEMP
      STOP
  370 IF(NGTOT.LE.MAXG) GO TO 380
      TYPE *,' NUMBER OF TARGETS AND GROUND STATIONS GREATER THAN'
      TYPE *,' PROGRAM IS DIMENSIONED FOR AT PRESENT.'
      ITEMP=MAXG
      TYPE *,' MAXIMUM NUMBER OF TARGETS AND GROUND STATIONS IS ',ITEMP
      STOP
  380 CONTINUE
      IF(INC.GT.0)GO TO 390
      TYPE *,' INC FOR DELAY MUST BE GREATER THAN 0, AND ',INC,'ISNT'
      STOP
  390 CONTINUE
      INP=0
C***
C***     SECTION TO SET UP GRID IN REST OF ELG(PART NOT USED BY
C***     TARINPUT, MOVETARG, AND SHIPTRAK)
C***     ON NEXT PASSES GRID WILL FINISH UP.
C***
      NGPP1=NGP+1
      IF (.NOT.GRID)GO TO 420
      FROG=MAX(SPALAT,1.)
      NUMLAT=1+ABS(BLAT-ELAT)/ABS(FROG)
      IF(BLAT.EQ.ELAT) NUMLAT=1
      FROG=MAX(SPALON,1.)
      NUMLON=1+ABS(BLON-ELON)/ABS(FROG)
      IF(BLON.EQ.ELON)NUMLON=1
      NUMTAR=NUMLAT*NUMLON
      DO 400 K=1,MAXGP
      IF(K.GT.NUMTAR) GO TO 410
```

```
      LASTK=K
      I=(K-1)/NUMLON+1
      J=MOD(K-1,NUMLON)+1
      L=K+NGTOT
      ELGS(L,1)=(I-1)*SPALAT+BLAT
      ELGS(L,2)=(J-1)*SPALON+BLON
      ELGS(L,3)=RADII
  400 ELGS(L,4)=2.05
  410 CONTINUE
      NGP=L-NGS
      NGTOT=NGP+NGS
  420 CONTINUE
      NGEND=NGP
      IF(SINGL)NGEND=1
      DO 540 INUM=1,NGEND
      NUMFIL=INUM+7
      OPEN(UNIT=NUMFIL,TYPE='NEW')
C***
C***     THIS WRITES TWO HEADERS, THE FIRST OF WHICH MAY BE TORN OFF
C***     THE SECOND HEADER WILL NOT CONTAIN GROUND STATION DATA
C***
      DO 530 IREDO=1,2
      WRITE(NUMFIL,800)
      IF(.NOT.SEC)WRITE(NUMFIL,570) DTT
      IF(SEC)WRITE(NUMFIL,580) DTT
      WRITE(NUMFIL,590) NSAT1,NSAT2,NGP,NGS
      WRITE(NUMFIL,760)
      WRITE(NUMFIL,600)
      WRITE(NUMFIL,610)ISY,ISM,ISD,ISH,ISMIN,SSEC
      WRITE(NUMFIL,620)ISTPY,ISTPM,ISTPD,ISTPH,ISTPMI,STPSEC
      WRITE(NUMFIL,760)
      IF(.NOT.DELA)GO TO 430
      WRITE(NUMFIL,750) TMINN,TMAXX
      WRITE(NUMFIL,860) INC
      WRITE(NUMFIL,760)
  430 CONTINUE
      WRITE(NUMFIL,760)
      WRITE(NUMFIL,630)
      WRITE(NUMFIL,640)
      INUMG=INUM+NGS
      IF(.NOT.SINGL)WRITE(NUMFIL,650)INUM,TYP(ELGS(INUMG,4)),
     1ELGS(INUMG,1),ELGS(INUMG,2)      ,ELGS(INUMG,3)
      IF(.NOT.SINGL) GO TO 490
      DO 480 I=NGSP1,NGP
      NUM=I-NGS
      GO TO (440,460,450,470)ELGS(I,4)
  440 WRITE(NUMFIL,650)NUM,TYP(ELGS(I,4)),ELGS(I,1),ELGS(I,2)
     1 ,ELGS(I,3)
      GO TO 480
  450 WRITE(NUMFIL,660)NUM,TYP(ELGS(I,4)),TRG(NUM,1),TRG(NUM,2),ZERO,
     1 TRG(NUM,3),TRG(NUM,4),TRG(NUM,5)
```

```
      GO TO 480
  460 GO TO 480
  470 WRITE(NUMFIL,670)NUM,TYP(ELGS(I,4)),TRG(NUM,1),TRG(NUM,2),ZERO,
     1  TRG(NUM,3),TRG(NUM,4)
  480 CONTINUE
      IF(GRID) WRITE(NUMFIL,850) BLAT,BLON,ELAT,ELON
     1 ,SPALAT,SPALON,RADII
  490 CONTINUE
      OUTFRC=FRAC*100.
      IF(MAT)WRITE(NUMFIL,890)PSTART,DELTAP,NUMP,OUTFRC
      IF(IREDO.EQ.2) GO TO 540
      IF (NGS.EQ.0) GO TO 510
      WRITE(NUMFIL,760)
      WRITE(NUMFIL,680)
      DO 500 IN=1,NGS
      WRITE(NUMFIL,690) IN,ELGS(IN,1),ELGS(IN,2),ELGS(IN,3),ELGS(IN,4)
  500 CONTINUE
  510 CONTINUE
      WRITE(NUMFIL,760)
      IF(VARIA)WRITE(NUMFIL,880) ITEM,START,AMOUNT,NUMVAR
      IARG=IRA
      IF(LONG)ARG=LON
      ARG2=MEANL
      IF(.NOT.MEAN)ARG2=TRUEL
      WRITE(NUMFIL,700)
      WRITE(NUMFIL,730)ARG2,ARG
      WRITE(NUMFIL,710)
      WRITE(NUMFIL,760)
      DO 520 I=1,NTOT
      IKS=I-NSAT1
      IK=I
      IF(I.GT.NSAT1)IK=IKS
      IF(VARIA.AND.I.GT.1) GO TO 530
      IFGS=ELSAT(I,11).EQ.1.0
  520 WRITE(NUMFIL,740) IK,(ELSAT(I,K),K=1,6),(IELSAT(I,J),J=1,5),
     1 SECSAT(I),IFGS,TYPA(ELSAT(I,12)),(ELSAT(I,K),K=14,15)
  530 WRITE(NUMFIL,810)
  540 CLOSE(UNIT=NUMFIL)
C***
C***   IF LOOPING ON GRIDS, EXIT HERE
C***
      IF(INP.GE.1) RETURN
      DO 560 I=1,NTOT
      ISATY=IELSAT(I,1)
      ISATM=IELSAT(I,2)
      ISATD=IELSAT(I,3)
      ISATH=IELSAT(I,4)
      ISATMI=IELSAT(I,5)
      ITEMP=IDATE(ISATY,ISATM,ISATD)
      TEMP=ITEMP
      TS=SECSAT(I)+60.*ISATMI+3600.*ISATH
```

```
      GHAN=GHA(TS,TEMP)
      ELSAT(I,10)=(ITEMP-ISTART)*86400.0+(TS-STSEC)
C***
C***     THIS CONVERTS R.A. TO LONG.
C***
      IF(.NOT.LONG)ELSAT(I,6)=ELSAT(I,6)-GHAN
      DO 550 K=3,6
  550 ELSAT(I,K)=ELSAT(I,K)*DTR
  560 CONTINUE
      IF(.NOT.SEC)TMIN=TMINN*60.
      IF(.NOT.SEC)TMAX=TMAXX*60
      IF(.NOT.SEC)DT=DTT*60.
      IF(SEC)DT=DTT
      IF(SEC)TMIN=TMINN
      IF(SEC)TMAX=TMAXX
      RETURN
  570 FORMAT(4X,'STEPSIZE(MIN)=',F8.2)
  580 FORMAT(4X,'STEPSIZE(SEC)=',F8.2)
  590 FORMAT(' # SYS1 SATS = ',I3,'  # SYS2 SATS = ',I3,
     1     '  # TARGETS = ',I3,'  # GROUND STATIONS = ',I3)
  600 FORMAT(15X,'YR  MO  DAY  HR  MIN     SEC')
  610 FORMAT(' START TIME IS ',3(I2,2X),1X,2(I2,2X),F8.2)
  620 FORMAT(' STOP TIME IS  ',3(I2,2X),1X,2(I2,2X),F8.2)
  630 FORMAT('                  TARGET        [DEG]          [DEG]
     1        [DEG]  ')
  640 FORMAT(' LOCATION       TYPE          LAT.         LONG.        '
     1  'RADIUS')
  650 FORMAT(' TARGET # ',I3,2X,A8,3(3X,F10.5))
  660 FORMAT(' TARGET # ',I3,2X,A8,3(3X,F10.5),' TO ',2(3X,F10.5),
     1  ' AT',F10.5,' KNOTS')
  670 FORMAT(' TARGET # ',I3,2X,A8,3(3X,F10.5),' TO ',2(3X,F10.5))
  680 FORMAT(' G. STATION                   LAT.         LONG.        '
     1  'RADIUS     ELEVATION')
  690 FORMAT(' G. STA. #',I3,10X,4(3X,F10.5))
  700 FORMAT(' SATELLITE ELEMENTS ',5X,'[DEG]',4X,'[DEG]',
     1 3X,'[DEG]',3X,'[DEG]',36X,'[DEG]',5X,'[DEG]')
  710 FORMAT('   #    A(KM)        E     INC       ANOM    PERI',
     1 '     NODE    YR  M  D  H  M  SEC GS   TYPE     PARAM1     PARAM2')
  720 FORMAT('                                                    ',
     1 '     R.A.                         TARG    STAT')
  730 FORMAT(35X,A4,11X,A4,23X,'IF',3X,'ANT.')
  740 FORMAT(1X,I3,1X,F11.4,1X,F7.5,1X,F7.4,1X,3(F7.3,1X),5(I2,1X),F5.2,
     1 1X,L1,3X,A4,2X,2(F8.3,2X))
  750 FORMAT(' MINIMUM DELAY(MIN)= ',F8.1,'  MAX DELAY(MIN)= ',F8.1)
  760 FORMAT(1H )
  770 FORMAT(' TYPE IN INPUT FILE NAME')
  780 FORMAT(40A1)
  790 FORMAT(' TYPE IN HEADER TITLE')
  800 FORMAT(1X,'                                                   ')
  810 FORMAT(1H1)
  820 FORMAT(' SYS1 SATELLITES DO NOT NEED TO SEE A GROUND STATION')
```

```
  830 FORMAT(A8)
  840 FORMAT(' THE WORD ',A8,' IS NOT ACCEPTABLE INPUT')
  850 FORMAT(' GRID FROM(',F8.3,',',F8.3,'),TO (',F8.3,',',F8.3,')'
     1,' BY (',F8.3,',',F8.3,') ,' , 'WITH RADII =',F9.2)
  860 FORMAT(' NUMBER OF SYS2 SAT REVISITS PER SYS1 SAT VISIT = ',I3)
  870 FORMAT(' MOVING TARGET FROM ',F8.3,',',F8.3,' TO ',F8.3,',',F8.3)
  880 FORMAT(' VARIATION OF SATELLITE ELEMENT ',I2,' FROM ',F8.3,' BY ',
     1 F8.3,' , ',I6,' TIMES')
  890 FORMAT('ØMATRIX GRID STARTS AT STEP ',I6,/,
     1 ' IS  ',I6,' STEPS LONG PER PERIOD,',/,
     2 ' FOR ',I6,' PERIODS, ',/,
     3 ' AND WITH AN ACCEPTABLE FRACTION OF ',F6.2,' PERCENT')
      END
      FUNCTION GHA(TSEC,TDAY)
C***
C******RETURN GREENWICH HOUR ANGLE FOR DATE FROM 1950
C******(IN DEGREES)
C***
      DATA RTD/57.29577951308232088E0/
      D=TDAY
      OMEGA=0.004178074216E0/(1.0+5.21E-13*D)
      DF=AMOD(0.985647346E0*D,360.0E0)
      TEMP=100.0755428E0+DF+2.9015E-13*D**2+OMEGA*TSEC
      DA=0.E0
      GHAN=AMOD(TEMP+DA*RTD,360.0E0)
      IF(GHAN.LT.0.0)GHAN=GHAN+360.0
      GHA=GHAN
      RETURN
      END
      FUNCTION IDATE(IYR,IMON,IDAY)
C***
C******IDATE RETURNS NUMBER OF DAYS FROM 1950
C******RANGE IS FROM 1950 TO 1999
C***
      DIMENSION MON(12),NYR(49)
      DATA MON/0,31,59,90,120,151,181,212,243,273,304,334/
      DATA NYR/0,365,730,1096,1461,1826,2191,2557,2922,3287,
     1 3652,4018,4383,4748,5113,5479,5844,6209,6574,6940,7305,
     2 7670,8035,8401,8766,9131,9496,9862,10227,10592,10957,
     3 11323,11688,12053,12418,12784,13149,13514,13879,14245,
     4 14610,14975,15340,15706,16071,16436,16801,17167,17532/
      IF(IYR.GT.49.AND.IYR.LT.100) GO TO 20
      IYR=IYR+1900
      TYPE 10,IYR
   10 FORMAT(' DATE OF ',I4,' IS OUT OF RANGE OF IDATE')
      STOP
   20 ISUM=0
      ISUM=ISUM+MON(IMON)
      L=MOD(IYR,4)
      IF(L.EQ.0.AND.IMON.GT.2)ISUM=ISUM+1
      ISUM=ISUM+IDAY
```

```
M=IYR-5Ø+1
IDATE=ISUM+NYR(M)-1
RETURN
END
```

8.3 Subroutine SETUP

```
C***
C***SETUP INITIALIZES AND DOES PRECALCULATIONS
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP= 4,MAXG= 8,MAXSAT= 4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      REAL*4 MANOM
C***  SKIP THE FIRST SECTION IF LOOPING ON A GRID
C***
      IF(INP.GE.1) GO TO 120
C***
C***  IF NECESSARY, CONVERT INPUT TRUE ANOMALY TO MEAN ANOMALY
C***
      IF(MEAN)GO TO 20
      DO 10 I=1,NTOT
      TANOM=ELSAT(I,4)
      ECC=ELSAT(I,2)
      EANOM=2*ATAN(SQRT((1-ECC)/(1+ECC))*TAN(TANOM/2.0))
      MANOM=EANOM-ECC*SIN(EANOM)
   10 ELSAT(I,4)=MANOM
   20 CONTINUE
C***
C***  COMPUTE SATELLITE CONSTANTS
C***
      FAC=XJ2*SQRT(XMU/(RE**3))
      DO 90 I=1,NTOT
      F=SQRT((RE/ELSAT(I,1))**7)
      F=F*FAC
```

```
      G=SQRT(XMU/(ELSAT(I,1)**3))
      CI=COS(ELSAT(I,3))
      D=(1.-ELSAT(I,2)*ELSAT(I,2))**2
      WDT=.75*F*(5.*CI*CI-1.)/D
      OMDT=-1.5*F*CI/D
      ELSAT(I,7)=WDT
      ELSAT(I,8)=OMDT
      ELSAT(I,9)=G
      ELSAT(I,13)=ELSAT(I,13)*DTR
      ELSAT(I,14)=ELSAT(I,14)*DTR
      ELSAT(I,15)=ELSAT(I,15)*DTR
      ELSAT(I,16)=ELSAT(I,16)*DTR
C***
C***   COMPUTE SATELLITE-ANTENNA CONSTANTS
C***
      GO TO(30,40,50,60,70,80)ELSAT(I,12)
C***
C***   ELEVATION ANGLE
C***
   30 COSE=COS(ELSAT(I,14))
      ELSAT(I,13)=COSE*RE
      GO TO 90
C***
C***  CO-DIP ANGLE
C***
   40 SINE=SIN(ELSAT(I,14))
      ELSAT(I,13)=SINE/RE
      GO TO 90
C***
C***   EARTH CENTRAL HALF-ANGLE
C***
   50 GO TO 90
C***
C***  SWATH HALF WIDTH IN N.MI.
C***
   60 GO TO 90
C***
C***  ANNULUS
C***
   70 ELSAT(I,13)=SIN(ELSAT(I,14))/RE
      GO TO 90
C***
C***  SIDE-LOOKING ANTENNA
C***
   80 ANGLE=ELSAT(I,14)+ELSAT(I,15)
      ELSAT(I,13)=SIN(ANGLE)/RE
      GO TO 90
   90 CONTINUE
      IF(NGS.EQ.0) GO TO 120
C***
C***   CONVERT LAT. AND LONG. TO XYZ EARTH-FIXED COORDINATES FOR
```

```
C***   GROUND STATIONS
C***
      DO 100 I=1,NGS
      ELG(I,1)=ELGS(I,1)*DTR
      ELG(I,2)=ELGS(I,2)*DTR
      ELG(I,4)=ELGS(I,3)*DTR
      ELG(I,5)=ELGS(I,4)*DTR
  100 CONTINUE
      DO 110 I=1,NGS
      X1=COS(ELG(I,1))*COS(ELG(I,2))
      X2=COS(ELG(I,1))*SIN(ELG(I,2))
      X3=SIN(ELG(I,1))
      ELG(I,1)=X1
      ELG(I,2)=X2
      ELG(I,3)=X3
  110 ELG(I,6)=COS(ELG(I,5))*RE
  120 CONTINUE
      DO 130 I=NGSP1,NGTOT
      ELG(I,1)=ELGS(I,1)*DTR
      ELG(I,2)=ELGS(I,2)*DTR
      ELG(I,4)=ELGS(I,3)*DTR
      ELG(I,5)=ELGS(I,4)
  130 CONTINUE
C***
C*** CONVERTS LAT. AND LONG. TO EARTH FIXED XYZ COORDINATES
C*** FOR TARGETS
C***
      DO 140 I=NGSP1,NGTOT
      X1=COS(ELG(I,1))*COS(ELG(I,2))
      X2=COS(ELG(I,1))*SIN(ELG(I,2))
      X3=SIN(ELG(I,1))
      ELG(I,1)=X1
      ELG(I,2)=X2
  140 ELG(I,3)=X3
C***
C*** R IS APOGEE RADIUS, USED FOR CALCULATING LARGEST GROUND COVERAGE
C*** HITAB WILL CONTAIN ARC OF LARGEST COVERAGE.  IF THERE IS A HIT,
C*** IT WILL COMPUTE COVERAGE EXACTLY
C*** HITAB IS A TABLE OF COSINES OF THE MAXIMUM ANGLE FOR ANY
C*** SATELLITE-GROUND POINT PAIR
C***
      DO 210 I=1,NTOT
      DO 210 J=1,NGP
      ISTOP(J)=LARGE
      L=J+NGS
      R=ELSAT(I,1)*(1+ELSAT(I,2))
      GO TO (150,160,170,180,190,200)ELSAT(I,12)
C*** ELEVATION ANGLE
C***
  150 PHI=ELSAT(I,13)/R
      HITAB(I,J)=COS(ACOS(PHI)-ELSAT(I,14)-ELG(L,4))
```

```
      GO TO 210
C***  CO-DIP ANGLE
C***
  160 PHI=MIN(1.0,ELSAT(I,13)*R)
      HITAB(I,J)=COS(ASIN(PHI)-ELSAT(I,14)-ELG(L,4))
      GO TO 210
C***  CENTRAL EARTH ANGLE
C***
  170 HITAB(I,J)=COS(ELSAT(I,14)-ELG(L,4))
      GO TO 210
C***   SWATH HALF-WIDTH
C***
  180 HITAB(I,J)=COS((ELSAT(I,14)/DTR)/RE-ELG(L,4))
      GO TO 210
C***   ANNULUS
C***
  190 PHI=ELSAT(I,13)*R
      HITAB(I,J)=COS(ASIN(PHI)-ELSAT(I,14)-ELG(L,4))
      GO TO 210
C***   SIDE VIEW ANTENNA
C***
  200 PHI=MIN(1.0,ELSAT(I,13)*R)
      HITAB(I,J)=COS(ASIN(PHI)-ELSAT(I,14)-ELG(L,4))
  210 CONTINUE
C***
C***   INITIALIZE BEGINNING OF TABLES
C***
      DO 220 I=1,MAXSAT
      SFIRST(I)=.TRUE.
      DO 220 J=1,MAXGP
      KNEXT(I,J)=1
      GS(I,1,J)=LARGE
      GS(I,2,J)=LARGE
  220 IFLAG(I,J)=0
      MAXGRS=MAXGP
      DO 230 I=1,MAXSAT
      DO 230 J=1,MAXGRS
  230 NUMGS(I,J)=15
      RETURN
      END
```

8.4 Subroutine COREL8

```
C***
C***      COREL8 CREATES THE INTERNAL WORK TABLE.
C***      THE TABLE IS ARRANGED BY SATELLITE AND TARGET AND CONTAINS
C***      THE START AND STOP TIMES FOR EACH SATELLITE DETECTION OF
C***      EACH TARGET.
C***      IT PROPAGATES EVERY SATELLITE AT THE SAME TIME.
C***      THIS GREATLY SIMPLIFIES THE ROUTINE, AT THE EXPENSE OF
C***      ITS DOING A LOT OF UNNECCESSARY WORK.  THE MORE TARGETS AND
C***      SATELLITES THERE ARE, THE SMALLER THE AMOUNT OF
C***      UNNECCESSARY WORK IT DOES, AND THE MORE EFFICIENT IT BECOMES.
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
C***      LOOP FOR EACH SATELLITE AT THE CURRENT TIME STEP
C***      NEWPOS UPDATES THE TARGET POSITIONS, IF NECESSARY
C***
      CALL NEWPOS
      DO 20 I=1,NTOT
      IK=I
C***
C***   ORBIT UPDATES SATELLITE POSITION AND DECIDES IF A HIT OCCURRED.
C***
      CALL ORBIT(IK)
C***
C***      LOOP OVER TARGETS FOR EACH SATELLITE
C***
      DO 20 J=1,NGP
```

```
      IF(IFDONE(J)) GO TO 20
      IF(IFHTGP(J).EQ.1) GO TO 10
C***
C***     NO HIT THIS TIME
C***
      IF(IFLAG(I,J).EQ.0) GO TO 20
C***
C***     THERE WAS A HIT LAST TIME
C***     THEREFOR WRITE DOWN STOP TIME OF HIT
C***
      IFLAG(I,J)=0
      IF(TIME/DT.GT.FLOAT(LARGE)) GO TO 30
      GS(I,KNEXT(I,J)+1,J)=TIME/DT
      IF(GS(I,KNEXT(I,J)+1,J).LT.0) GO TO 30
      KNEXT(I,J)=KNEXT(I,J)+2
      GS(I,KNEXT(I,J),J)=LARGE
      GS(I,KNEXT(I,J)+1,J)=LARGE
      IF(KNEXT(I,J).LT.MAXGS) GO TO 20
      NUMDT=DT/60.0
      TOUT=TIME/60.
      TYPE 40,TOUT
      STOP
C***
C***     THE FOLLOWING IS IF THERE IS A HIT THIS TIME
C***
   10 IF(IFLAG(I,J).EQ.1) GO TO 20
C***
C***     THERE WASN'T A HIT LAST TIME SO WRITE DOWN START TIME OF HIT
C***
      IF(TIME/DT.GT.FLOAT(LARGE)) GO TO 30
      IFLAG(I,J)=1
      GS(I,KNEXT(I,J),J)=TIME/DT
      GS(I,KNEXT(I,J)+1,J)=LARGE
      IF(GS(I,KNEXT(I,J),J).LT.0) GO TO 30
      IF(ELSAT(I,11).NE.1.0) GO TO 20
C***
C***     WRITE DOWN IN IGS THE GROUND STATION
C***
      KNO2=KNEXT(I,J)/2+1
      IGS(I,KNO2,J)=NUMGS(I,J)
   20 CONTINUE
      RETURN
   30 TOUT=TIME/60.
      TYPE 50,TOUT
      TIME=TSTOP+2*DT
      RETURN
   40 FORMAT(' STORAGE OVERFLOW IN SUB COREL8 AT TIME(MIN) ',F8.0)
   50 FORMAT(' RUN TIME TOO LONG IN SUB COREL8 ',/,
     1 ' INTEGER TIME STEPS HAVE GONE NEGATIVE '
     2 ,/,' TIME IS ',F9.0)
      END
```

8.5 Subroutine ORBIT (ISAT)

```
C***
C***     ORBIT IS THE EPHEMERIS GENERATOR DRIVER.
C***     IT IS CALLED FOR A PARTICULAR
C***     SATELLITE BUT DOES ALL TARGETS FOR THAT SATELLITE.
C***   IT CHECKS  GROUND STATION VISIBILITY
C***   ORBIT RETURNS A YES FOR A DETECTION AS A ONE(1)
C***   FOR EACH TARGET IN IGS
C***   AND RETURNS THE GROUND STATION NUMBER IN NUMGS.
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      REAL LATB(2),LONB(2),LATBN(MAXSAT),LONBN(MAXSAT)
     1    ,LATBF(MAXSAT),LONBF(MAXSAT),LATEN(MAXSAT),LONEN(MAXSAT)
     2    ,LATARG,LOTARG,LATEF(MAXSAT),LONEF(MAXSAT),DIPP(2)
      LOGICAL HIT
      INTEGER COUNT(MAXSAT)
C***
      DIMENSION EL(6),ELT(6),R(6),A(3),B(3),D(3)
C***
C***
      COUNT(ISAT)=0
      DO 10 I=1,6
   10 EL(I)=ELSAT(ISAT,I)
      WDT=ELSAT(ISAT,7)
      OMDT=ELSAT(ISAT,8)
      XN=ELSAT(ISAT,9)
      TIM=TIME-ELSAT(ISAT,10)
```

```
C***
C***     TO USE A DIFFERENT EPHEMERIS GENERATOR JUST REPLACE PREDIC, AND
C***     IF NECESSARY CHANGE THE CALL FORMAT
C***
      CALL PREDIC(EL,TIM,XN,WDT,OMDT,R,RMAG)
   20 DO 340 J=1,NGP
C***
C***     FOR THE SATELLITE, CHECK FOR A DETECTION FOR EACH TARGET
C***
      K=J+NGS
      IFHTGP(J)=0
C***
C***     SET THE DETECTION FLAG TO 0
C***
   30 DOT=0
      DO 40 I=1,3
   40 DOT=DOT+R(I)*ELG(K,I)
C***
C***     THE FOLLOWING IS A QUICK TEST-IF IT FAILS GO ON TO NEXT TARGET
C***
      IF(DOT.LE.HITAB(ISAT,J)) GO TO 340
C***
C***     PASSED QUICK TEST SO DO EXACT CALCULATION
C***
      GO TO (50,60,70,80,90,100)ELSAT(ISAT,12)
C***
C***   ELEVATION ANGLE
C***
   50 COSPHI=COS(ACOS(ELSAT(ISAT,13)/RMAG)-ELSAT(ISAT,14)-ELG(K,4))
      IF(DOT.LE.COSPHI) GO TO 340
      GO TO 290
C***
C***  CONE
C***
   60 COSPHI=COS(ASIN(ELSAT(ISAT,13)*RMAG)-ELSAT(ISAT,14)-ELG(K,4))
      IF(DOT.LE.COSPHI)GO TO 340
      GO TO 290
C***
C***  CENT
C***
   70 GO TO 290
C***
C***  DIST
C***
   80 GO TO 290
C***
C*** ANNULUS
C***
   90 COSPH1=COS(ASIN(ELSAT(ISAT,13)*RMAG)-ELSAT(ISAT,14)-ELG(K,4))
      COSPH2=COS(ASIN((SIN(ELSAT(ISAT,15))/RE)*RMAG)-ELSAT(ISAT,15)
     1   -ELG(K,4))
```

```
      IF(DOT.LE.COSPH1.AND.DOT.GT.COSPH2)GO TO 340
      GO TO 290
C***
C*** SIDE
C***
C*** ELSAT 14 IS THE CO-DIP, ELSAT 15 IS THE BEAM WIDTH
C*** FIRST CALCULATE THE NEAR AND FAR POINTS OF THE SWATH
C*** TRACED OUT BY DIP+BEAM/2, AND DIP-BEAM/2
C***
  100 IF(COUNT(ISAT).GT.0.AND.SFIRST(ISAT))GO TO 340
      IF(COUNT(ISAT).GT.0) GO TO 130
      COUNT(ISAT)=COUNT(ISAT)+1
      DIPP(1)=PI/2.-ELSAT(ISAT,14)-ELSAT(ISAT,15)/2.
      DIPP(2)=PI/2.-ELSAT(ISAT,14)+ELSAT(ISAT,15)/2.
C***
C*** R WAS NORMALLIZED, SO UN-NORMALLIZE IT.
C***
      R(1)=R(1)*RMAG
      R(2)=R(2)*RMAG
      R(3)=R(3)*RMAG
C***
C*** A IS R CROSS V
C***
      A(1)=R(2)*R(6)-R(3)*R(5)
      A(2)=R(3)*R(4)-R(1)*R(6)
      A(3)=R(1)*R(5)-R(2)*R(4)
      ANORM=SQRT(A(1)**2+A(2)**2+A(3)**2)
C***
C*** NORMALLIZED, THAT IS
C***
      A(1)=A(1)/ANORM
      A(2)=A(2)/ANORM
      A(3)=A(3)/ANORM
      DO 110 ICOUNT=1,2
      DIP=DIPP(ICOUNT)
      COSDIP=COS(DIP)
      SINDIP=SIN(DIP)
C***
C*** D IS A ROTATED ABOUT V BY THE DIP ANGLE
C***
      D(1)=A(1)*COSDIP-R(1)*SINDIP/RMAG
      D(2)=A(2)*COSDIP-R(2)*SINDIP/RMAG
      D(3)=A(3)*COSDIP-R(3)*SINDIP/RMAG
      ANORM=SQRT(A(1)**2+A(2)**2+A(3)**2)
      DNORM=SQRT(D(1)**2+D(2)**2+D(3)**2)
C***
C*** B1 IS THE ANGLE BETWEEN D AND R
C***
C***
C*** B IS THE CO-DIP ANGLE
C***
```

```
      ANGLEB=PI/2.-DIP
C***
C*** R IS ANGLE OPPOSITE VECTOR R
C***
      ANGLER=ACOS((RMAG*SIN(ANGLEB))/RE)+PI/2
C***
C*** D IS THIRD ANGLE IN TRIANGLE, ANGLE FROM R TO GROUND POINT
C***
      ANGLED=PI-ANGLEB-ANGLER
      COSAND=COS(ANGLED)
      SINAND=SIN(ANGLED)
C***
C*** VECTOR B IS R ROTATED ABOUT V TOWARD A BY ANGLE D
C***
      B(1)=R(1)*COSAND/RMAG+A(1)*SINAND
      B(2)=R(2)*COSAND/RMAG+A(2)*SINAND
      B(3)=R(3)*COSAND/RMAG+A(3)*SINAND
      DNORM2=RMAG*SIN(ANGLED)/SIN(ANGLER)
      BNORM1=SQRT(B(1)**2+B(2)**2)
      LATB(ICOUNT)=ATAN(B(3)/BNORM1)*RTD
      LONB(ICOUNT)=ARKTNS(360,B(1),B(2))*RTD
  110 CONTINUE
      RMAG2=SQRT(R(1)**2+R(2)**2)
      SATLAT=ATAN(R(3)/RMAG2)*RTD
      SATLON=ARKTNS(360,R(1),R(2))*RTD
D     TYPE *,' SAT LAT, LON = ',SATLAT,SATLON
D     TYPE *,' ANTENNA LAT,LON = ', LATB(1),LONB(1)
D     TYPE *,' ANTENNA LAT,LON = ', LATB(2),LONB(2)
      IF(.NOT.SFIRST(ISAT)) GO TO 120
      LATBF(ISAT)=LATB(1)
      LONBF(ISAT)=LONB(1)
      LATBN(ISAT)=LATB(2)
      LONBN(ISAT)=LONB(2)
      LATEF(ISAT)=LATB(1)
      LONEF(ISAT)=LONB(1)
      LATEN(ISAT)=LATB(2)
      LONEN(ISAT)=LONB(2)
      GO TO 340
  120 LATEF(ISAT)=LATB(1)
      LONEF(ISAT)=LONB(1)
      LATEN(ISAT)=LATB(2)
      LONEN(ISAT)=LONB(2)
D     TYPE *,' FIRST POINTS ARE'
D     TYPE *, LATBF(ISAT),LONBF(ISAT),LATBN(ISAT),LONBN(ISAT)
D     TYPE *,' NEXT POINTS ARE'
D     TYPE *, LATEF(ISAT),LONEF(ISAT),LATEN(ISAT),LONEN(ISAT)
C***
C***    NOW HAVE RECTANGLE ON EARTH DEFINED BY FOUR POINTS.
C***    THE POINTS ARE DEFINED AS (B)EGIN, (E)ND, (N)EAR, AND (F)AR
C***    NOW HAVE TO DETERMINE IF TARGET POINT IS INSIDE THE
C***    RECTANGLE.
```

```
C***
  130 CONTINUE
C***
C*** TEST SECTION
C***
C*** FIRST HAVE TO GET POINTS CLOSE TO ONE ANOTHER ON THE
C*** COORDINATE SYSTEM, IE., 5 DEG LONG AND 355 LONG ARE
C*** REALLY ONLY 10 DEG. APART.
      IF(ABS(LONEF(ISAT)-LONBN(ISAT)).LT.180.)GO TO 140
      LONBN(ISAT)=LONBN(ISAT)+SIGN(360.,(LONEF(ISAT)-LONBN(ISAT)))
  140 IF(ABS(LONEF(ISAT)-LONEN(ISAT)).LT.180.)GO TO 150
      LONEN(ISAT)=LONEN(ISAT)+SIGN(360.,(LONEF(ISAT)-LONEN(ISAT)))
  150 IF(ABS(LONEF(ISAT)-LONBF(ISAT)).LT.180.)GO TO 160
      LONBF(ISAT)=LONBF(ISAT)+SIGN(360.,(LONEF(ISAT)-LONBF(ISAT)))
  160 CONTINUE
      LATARG=ASIN(ELG(J,3))*RTD
      LOTARG=ARKTNS(360,ELG(J,1),ELG(J,2))*RTD
D     TYPE *,' ELG(J,3)= ',ELG(J,3)
      IF(ABS(LONEF(ISAT)-LOTARG).LT.180.) GO TO 170
      LOTARG=LOTARG+SIGN(360.,(LONEF(ISAT)-LOTARG))
  170 CONTINUE
      IF(LONEF(ISAT).NE.LONBF(ISAT)) GO TO 180
      IF(LOTARG.LT.LONEF(ISAT)) GO TO 260
      GO TO 190
  180 SLOPET=(LATEF(ISAT)-LATBF(ISAT))/(LONEF(ISAT)-LONBF(ISAT))
      YTOP=SLOPET*LOTARG+(LATEF(ISAT)-SLOPET*LONEF(ISAT))
      IF(LATARG.GT.YTOP)GO TO 260
  190 IF(LONEN(ISAT).NE.LONBN(ISAT)) GO TO 200
      IF(LOTARG.GT.LONEN(ISAT)) GO TO 260
      GO TO 210
  200 SLOPEB=(LATEN(ISAT)-LATBN(ISAT))/(LONEN(ISAT)-LONBN(ISAT))
      YBOT=SLOPEB*LOTARG+(LATEN(ISAT)-SLOPEB*LONEN(ISAT))
      IF(LATARG.LT.YBOT) GO TO 260
  210 IF(LONBF(ISAT).NE.LONBN(ISAT)) GO TO 220
      IF(LOTARG.LT.LONBF(ISAT)) GO TO 260
      GO TO 230
  220 SLOPEL=(LATBF(ISAT)-LATBN(ISAT))/(LONBF(ISAT)-LONBN(ISAT))
      YLEFT=SLOPEL*LOTARG+(LATBF(ISAT)-SLOPEL*LONBF(ISAT))
      IF(SLOPEL.LT.0.AND.LATARG.LT.YLEFT) GO TO 260
      IF(SLOPEL.GE.0.AND.LATARG.GT.YLEFT) GO TO 260
  230 IF(LONEF(ISAT).NE.LONEN(ISAT)) GO TO 240
      IF(LOTARG.GT.LONEF(ISAT)) GO TO 260
      GO TO 250
  240 SLOPER=(LATEF(ISAT)-LATEN(ISAT))/(LONEF(ISAT)-LONEN(ISAT))
      YRIGHT=SLOPER*LOTARG+(LATEF(ISAT)-SLOPER*LONEF(ISAT))
      IF(SLOPER.LT.0.AND.LATARG.GT.YRIGHT) GO TO 260
      IF(SLOPER.GE.0.AND.LATARG.LT.YRIGHT) GO TO 260
  250 CONTINUE
C***
C*** BINGO!                    WE HAVE A HIT
C***
```

```
      HIT=.TRUE.
D     TYPE *,' HIT OCCURRED AT TIME STEP ',TIME
D     TYPE *,' WITH TARGET ',J,' AT LAT ',LATARG,', LON ',LOTARG
D     TYPE *,' WHICH WAS BETWEEN THE POINTS '
D     TYPE *, LATBF(ISAT),LONBF(ISAT)
D     TYPE *, LATEF(ISAT),LONEF(ISAT)
D     TYPE *, LATBN(ISAT),LONBN(ISAT)
D     TYPE *, LATEN(ISAT),LONEN(ISAT)
      GO TO 270
  260 CONTINUE
D     TYPE *,' NO HIT, POINTS WERE: TARGET= ',LATARG,LOTARG
D     TYPE *, LATBF(ISAT),LONBF(ISAT)
D     TYPE *, LATEF(ISAT),LONEF(ISAT)
D     TYPE *, LATBN(ISAT),LONBN(ISAT)
D     TYPE *, LATEN(ISAT),LONEN(ISAT)
      HIT=.FALSE.
  270 CONTINUE
      IF(.NOT.HIT) GO TO 340
  280 CONTINUE
C***
C***      PASSED SECOND TEST-COUNT IT AS A DETECTION
C***
  290 IFHTGP(J)=1
      IF(ELSAT(ISAT,11).NE.1.0) GO TO 340
C***
C***     FOR SATELLITES THAT REQUIRE A GROUND STATION, CHECK FOR
C***     DETECTION OF A GROUND STATION.
C***
  300 CONTINUE
      DO 320 L=1,NGS
      COSPH2=COS(ACOS(ELG(L,6)/RMAG)-ELG(L,5)-ELG(L,4))
      DOT=0.
      NGSTA=L
      DO 310 I=1,3
  310 DOT=DOT+R(I)*ELG(L,I)
  320 IF(DOT.GT.COSPH2) GO TO 330
C***
C***     NO GROUND STATION VISIBLE SO CAN'T COUNT THE DETECTION
C***
      IFHTGP(J)=0
  330 CONTINUE
      NUMGS(ISAT,J)=NGSTA
  340 CONTINUE
C***
C*** RESET BEGINNING POINTS FOR SIDE-LOOKING ANTENNA
C***
      LATBF(ISAT)=LATEF(ISAT)
      LONBF(ISAT)=LONEF(ISAT)
      LATBN(ISAT)=LATEN(ISAT)
      LONBN(ISAT)=LONEN(ISAT)
      IF(COUNT(ISAT).GT.0)SFIRST(ISAT)=.FALSE.
```

```
      IF(COUNT(ISAT).EQ.Ø)SFIRST(ISAT)=.TRUE.
      RETURN
      END
```

8.6 Subroutine PREDIC (EL,T,XN,WDT,OMDT,R,RMAG)

```
C***
C***     THIS IS THE ANALYTIC EPHEMERIS GENERATOR
C***
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
C***
      DIMENSION R(6),EL(6),ELT(6)
C***
C***      T IS TIME FROM EPOCH IN SECONDS
C***      EL IS IN KM AND RADIANS
C*** COMPUTE XL AT TIME T
C***
      XL=XN*T+EL(4)
      XL=MOD(XL,TPI)
      IF(XL.LT.Ø.)XL=XL+TPI
      ELT(4)=XL
C***
C*** COMPUTE RATES (SECULAR)
C*** COMPUTE RA OF NODE AND ARG OF PERIGEE
C***
      OM=EL(6)+OMDT*T-WE*T
      W=EL(5)+WDT*T
      ELT(6)=MOD(OM,TPI)
      IF(ELT(6).LT.Ø.) ELT(6)=ELT(6)+TPI
      ELT(5)=MOD(W,TPI)
      IF(ELT(5).LT.Ø.) ELT(5)=ELT(5)+TPI
      DO 1Ø I=1,3
   1Ø ELT(I)=EL(I)
C***
C*** INPUT IS KM AND RADIANS AND MEAN ANOMALY
C*** OUTPUT IS XYZ COORD. AND MAG.
C*** COMPUTE TRUE ANOMALY
C***
      TOL=2.E-6
      ECC=ELT(2)
      XM=ELT(4)
C***
C***  SOLVES KEPLER'S EQUATION
C***  NOTE THAT THE TOLERANCE SENT TO IT IS DEPENDANT ON
C***  THE PRECISION AND WORD SIZE OF THE COMPUTER
C***
      EOLD=XM
      DO 2Ø K=1,1ØØ
      SEC=SIN(EOLD)*ECC
      CEC=COS(EOLD)*ECC
      ENEW=(XM+SEC-EOLD*CEC)/(1.EØ-CEC)
      DE=ABS(ENEW-EOLD)
      IF(DE.LE.TOL) GO TO 3Ø
```

```
   20 EOLD=ENEW
      TYPE 40
      STOP
   30 XKEP=ENEW
   40 FORMAT(10X,' **KEPLERS EQN DID NOT CONVERGE***')
      E=XKEP/2.E0
      TA=2.E0*ATAN(SQRT((1.E0+ELT(2))/(1.E0-ELT(2)))*SIN(E)/COS(E))
      RMAG=ELT(1)*(1.E0-ELT(2)*ELT(2))/(1.E0+ELT(2)*COS(TA))
      S=TA+ELT(5)
      SS=SIN(S)
      CS=COS(S)
      SO=SIN(ELT(6))
      CO=COS(ELT(6))
      SI=SIN(ELT(3))
      CI=COS(ELT(3))
      R(1)=CO*CS-SO*CI*SS
      R(2)=SO*CS+CO*CI*SS
      R(3)=SI*SS
C***
C***    THE FOLLOWING IS INCLUDED TO CALCULATE THE VELOCITY VECTOR
C***    IF NEEDED.
C***
      VR=SQRT(XMU/(RE*(1-ECC**2)))*ECC*SIN(TA)
      VT=SQRT(XMU*ELT(1)*(1-ECC**2))/RMAG
      R(4)=VR*(CO*CS-SO*SS*CI)+VT*(-CO*SS-SO*CI*CS)
      R(5)=VR*(SO*CS+CO*SS*CI)+VT*(-SO*SS+CO*CI*CS)
      R(6)=VR*(SI*SS)+VT*(SI*CS)
C       RDMAG=SQRT(RDOT(1)**2+RDOT(2)**2+RDOT(3)**2)
      RETURN
      END
```

8.7 Subroutine NEWPOS

```
C***
C*** SUBROUTINE TO CALCULATE NEW POSITIONS FOR TARGETS
C***     IT WILL WORK FOR STATIONARY, MOVING, AND SHIP TRACKS
C***     (GREAT CIRCLES)
C***    MODIFIED TO DO ALL AT SAME TIME
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
```

```
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      LOGICAL START(MAXGP)
C***
      DATA START/MAXGP*.TRUE./
C***
      IFDON=0
      DO 100 I=1,NGP
      ICALL=I+NGS
      GO TO (10,20,30,50)ELG(ICALL,5)
C***    OOPS
      GO TO 80
C***
C***    TARINPUT TARGET
C***
   10 GO TO 100
C***
C***    TARGRID TARGET
C***
   20 GO TO 100
C***
C***   SHIPTRAK TARGET
C***
   30 IF(.NOT.START(I)) GO TO 40
      START(I)=.FALSE.
      CALL TRACK(TRG(I,1),TRG(I,2),TRG(I,3),TRG(I,4),I,TRG(I,5))
      GO TO 100
   40 CALL SHPMOV(I)
      GO TO 100
C***
C***    MOVETARG TARGET
C***
   50 MOVT=I
C***   SECTION FOR MOVING TARGETS.
C***   IT READS THE NEW XYZ EARTH-FIXED COORDINATES FOR EACH TARGET
C***   AT EACH TIME STEP.  TARGET FILES ARE CREATED BY PROGRAM TARGET.
C***   IF THE FILE ENDS BEFORE THE PROGRAM WANTS TO THE PROGRAM
C***   ENDS CORLE8 AND STARTS POST PROCESSING, IE., WHEN THE
C***   SHIP REACHES PORT, QUIT MOVING THE SATELLITES.
C***
      IF(IFDONE(MOVT))GO TO 70
      MOVFIL=100-MOVT
```

```
      MOVTRG=NGS+MOVT
      READ(MOVFIL,END=60)IT,ELG(MOVTRG,1),ELG(MOVTRG,2),ELG(MOVTRG,3)
      GO TO 70
   60 IFDONE(MOVT)=.TRUE.
      IFDON=IFDON+1
      IF(IFDON.EQ.NGP)ALLDUN=.TRUE.
      ISTOP(MOVT)=TIME/DT
   70 CONTINUE
      GO TO 100
   80 TYPE 90
   90 FORMAT(1X,' WRONG OPTION TO NEWPOS')
      STOP
  100 CONTINUE
      RETURN
      END
```

8.8 Subroutine TRACK (BLAT1,BLON1,ELAT1,ELON1,NIM,SPEED)

```
C***
C*** SUBROUTINE TO CREATE GREAT CIRCLE SHIP TRACKS
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
C***
      DIMENSION X1(MAXGP),X2(MAXGP),X3(MAXGP),X4(MAXGP)
      DIMENSION Y1(MAXGP),Y2(MAXGP),Y3(MAXGP),Y4(MAXGP)
      DIMENSION Z1(MAXGP),Z2(MAXGP),Z3(MAXGP),Z4(MAXGP)
      DIMENSION THETA(MAXGP),TOTIME(MAXGP),SPED(MAXGP)
C***
C***
```

```
      BLAT=BLAT1*DTR
      BLON=BLON1*DTR
      ELAT=ELAT1*DTR
      ELON=ELON1*DTR
      SPED(NIM)=SPEED
      IFDON=Ø
C***
C*** 1 REFERS TO VECTOR OF STARTING POINT
C*** 2 REFERS TO VECTOR OF ENDING POINT
C*** 3 REFERS TO VECTOR BINORMAL TO 1 AND 2
C*** 4 REFERS TO VECTOR BINORMAL TO 1 AND 3
C*** NEW REFERS TO OUTPUT VECTOR, 1 ROTATED BY THETA ABOUT 3
C***
      X1(NIM)=COS(BLAT)*COS(BLON)
      Y1(NIM)=COS(BLAT)*SIN(BLON)
      Z1(NIM)=SIN(BLAT)
      X2(NIM)=COS(ELAT)*COS(ELON)
      Y2(NIM)=COS(ELAT)*SIN(ELON)
      Z2(NIM)=SIN(ELAT)
      X3(NIM)=Y1(NIM)*Z2(NIM)-Z1(NIM)*Y2(NIM)
      Y3(NIM)=Z1(NIM)*X2(NIM)-Z2(NIM)*X1(NIM)
      Z3(NIM)=X1(NIM)*Y2(NIM)-X2(NIM)*Y1(NIM)
C***
C*** NORMALIZE VECTOR 3 AS 1 AND 2 ARE NORMALIZED BUT NOT
C*** NECESSARILY NORMAL TO EACH OTHER.
C***
      IF(X3(NIM)**2+Y3(NIM)**2+Z3(NIM)**2.NE.Ø.) GO TO 1Ø
      X3(NIM)=Ø.
      Y3(NIM)=Ø.
      Z3(NIM)=Ø.
      GO TO 2Ø
   1Ø CONTINUE
      THRNRM=SQRT(X3(NIM)**2+Y3(NIM)**2+Z3(NIM)**2)
      X3(NIM)=X3(NIM)/THRNRM
      Y3(NIM)=Y3(NIM)/THRNRM
      Z3(NIM)=Z3(NIM)/THRNRM
   2Ø X4(NIM)=Z1(NIM)*Y3(NIM)-Y1(NIM)*Z3(NIM)
      Y4(NIM)=X1(NIM)*Z3(NIM)-Z1(NIM)*X3(NIM)
      Z4(NIM)=Y1(NIM)*X3(NIM)-X1(NIM)*Y3(NIM)
C***
C*** THETA IS THE ANGLE BETWEEN 1 AND 2
C*** WE WILL STEP BY DELTHET THROUGH THETA
C*** FROM 1 TO 2 ALONG THE GREAT CIRCLE
C***
      CTHETA=X1(NIM)*X2(NIM)+Y1(NIM)*Y2(NIM)+Z1(NIM)*Z2(NIM)
      THETA(NIM)=ACOS(CTHETA)
C***
C*** THIS SECTION DETERMINES THE SIZE OF DELTHET AND THEREFOR
C*** THE NUMBER OF STEPS, NUMT.
C*** THERE ARE TWO WAYS OF DOING THIS: BY SPECIFYING THE LENGTH OF
C*** TIME AND THE TIME STEP, OR BY SPECIFYING THE SPEED OF THE SHIP
```

```
C*** AND THE TIME STEP AND LET THE PROGRAM FIGURE OUT HOW LONG IT
C*** TAKES TO GET THERE AT THAT SPEED.
C***
      IF(SPEED.EQ.0.) GO TO 40
   30 TOTIME(NIM)=60.*(THETA(NIM)/DTR)/(SPEED/3600.)
   40 CONTINUE
C***
      RETURN
C*** AT THIS POINT WE STEP THROUGH THETA BY DELTHET
C*** THIS METHOD DOES NOT USE ANY SMALL ANGLE APPROXIMATIONS
C*** RATHER THAN ITERATE HERE, AND WRITE THE RESULTS OUT TO A FILE,
C*** ONE COULD USE THIS AS A SUBROUTINE WITH THE ABOVE
C*** AS INITIALIZATION AND THE BELOW WITHOUT THE DO LOOP AS AN
C*** INDIVIDUAL CALL.
C***
      ENTRY SHPMOV(NUM)
      IF(SPED(NUM).EQ.0.)TOTIME(NUM)=TSTOP
      NOM=NGS+NUM
      IF(TIME.EQ.0.)RETURN
      IF(TIME.LE.TOTIME(NUM))GO TO 50
      IFDONE(NUM)=.TRUE.
      IFDON=IFDON+1
      IF(IFDON.GE.NGP)ALLDUN=.TRUE.
      ISTOP(NUM)=TIME/DT
      GO TO 60
   50 CONTINUE
      THET=-(TIME/TOTIME(NUM))*THETA(NUM)
      CTHETA=COS(THET)
      STHETA=SIN(THET)
      ELG(NOM,1)=X1(NUM)*CTHETA-X4(NUM)*STHETA
      ELG(NOM,2)=Y1(NUM)*CTHETA-Y4(NUM)*STHETA
      ELG(NOM,3)=Z1(NUM)*CTHETA-Z4(NUM)*STHETA
   60 CONTINUE
      RETURN
      END
      FUNCTION ARKTNS(N,X,Y)
C     COMPUTES 4-QUADRANT ARCTANGENT OF Y/X IN RADIANS
C     N=360  ANGLE LIES IN RANGE (0,360) DEG
C     N=180  ANGLE LIES IN RANGE (-180,180) DEG
      TPI = 6.283185307179586480D0
      IF(X.NE.0.D0) GO TO 10
      ARKTNS=SIGN(TPI/4.0,Y)
      IF(N.EQ.360.AND.ARKTNS.LT.0.) ARKTNS=TPI+ARKTNS
      RETURN
   10 F=ATAN2(Y,X)
      IF(N.EQ.360.AND.F.LT.0.D0)F=TPI+F
      ARKTNS=F
      RETURN
      END
```

8.9 *Subroutine* DELAY

```
C***
C***  DELAY IS THE DELAY POST PROCESSOR
C***  IT TAKES THE "GS" FILE CREATED BY  COREL8 AND ANALYZES THE
C***  DATA TO FIND THE DELAY TIMES FOR SYS1 PASSES FOLLOWED
C***  BY SYS2 PASSES.
C***  THE FLAG INC DETERMINES THE NUMBER OF REVISITS OF THE SYS2
C***  SYSTEM AFTER
C***  A SYS 1 HIT.  THE OUTPUT FILE IS SORTED BY SYS1 HIT.
C***  ARRAY "GS" IS AN ARRAY OF TIME STEPS FOR EACH SATELLITE.
C***  ITS UNITS OF TIME
C***  ARE INTEGER INPUT TIME STEPS(IE., FOR AN INPUT DT OF 120.
C***  (2 MINUTES), THE ARRAY ELEMENT WITH A VALUE OF 20 REPRESENTS
C***  20 TIME
C***  STEPS OR 40 MINUTES FROM TSTART).
C***  THE TIMES FOR EACH SATELLITE ARE THE TIME STEP THAT THE SATELLITE
C***  STARTS SEEING THE TARGET AND THE TIME STEP THAT IT FIRST DOES NOT
C***  SEE THE SATELLITE.
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      INTEGER FIRST1,FIRST2,IFSEE,NEXT1,NEXT2,LAST2
      INTEGER ITMIN,ITMAX,ITIMD,IT1,IT2,IDIFF
C***
      DIMENSION NEXT1(MAXSAT),NEXT2(MAXSAT),LAST2(MAXSAT)
C***
      ITMIN=TMIN/DT
      ITMAX=TMAX/DT
```

```
      DO 10 I=1,20
      DIFMAX(I)=0.
      DIFMIN(I)=1.E32
      DIFTT(I)=0.
      NUMDF(I)=0
   10 CONTINUE
      DIFFMX=0.
      DIFFMN=1.E33
      DIFTOT=0.
      NUMDIF=0
      DO 20 I=1,MAXSAT
   20 NEXT1(I)=1
      DO 30 I=1,MAXSAT
      NEXT2(I)=1
   30 LAST2(I)=1
      KMAX=100
      FIRST1=1
      FIRST2=N1P1
C***
C***           MAXSEE IS NUMBER OF REVISITS, IFSEE IS COUNTER
C***
      MAXSEE=INC
   40 IFSEE=0
C***
C***           FIRST FIND FIRST HIT OF SYS1
C***
      DO 50 I=1,NSAT1
   50 IF(GS(I,NEXT1(I),NUMGP).LT.GS(FIRST1,NEXT1(FIRST1),NUMGP))
     1 FIRST1=I
C***
C***        IF THE NEXT SYS1 HIT IS PAST BIGNUM, WE'RE THROUGH
C***
      IF(NEXT1(FIRST1).GE.KNEXT(FIRST1,NUMGP)) RETURN
C***
C***        SET SYS2 SATS. TO FIRST THAT CAN BE SEEN
C***
      DO 60 I=N1P1,NTOT
   60 NEXT2(I)=LAST2(I)
C***
C***        NOW FIND FIRST SYS2 SAT HIT OF THE ALLOWED GROUP
C***
   70 DO 80 I=N1P1,NTOT
   80 IF(GS(I,NEXT2(I),NUMGP).LT.GS(FIRST2,NEXT2(FIRST2),NUMGP))
     1 FIRST2=I
C***
C***        IF THE SYS1 START TIME IS GREATER THAN THE SYS2 STOP TIME,
C***        THE SYS2 TIME IS OBVIOUSLY TOO EARLY, SO INCREMENT
C***        THE SYS2'S SO IT WON'T BE CHECKED ANYMORE,AND START OVER
C***
   90 IF(NEXT2(FIRST2).GE.KNEXT(FIRST2,NUMGP)) GO TO 120
      IF(GS(FIRST1,NEXT1(FIRST1),NUMGP).LT.GS(FIRST2,NEXT2(FIRST2)+1,
```

```
     1 NUMGP))  GO TO 100
      NEXT2(FIRST2)=NEXT2(FIRST2)+2
      LAST2(FIRST2)=NEXT2(FIRST2)
      GO TO 70
C***
C***        TIMD IS THE DELAY. IF THE DELAY IS ITMAX, THE TIMES WILL
C***        BE THE SYS2 TIME AND (SYS2-ITMAX), WITH THE DELAY ITMAX.
C***        IF THE DELAY IS LESS THAN ITMIN, IT CHECKS TO SEE IF
C***        DURING THE CURRENT PASS IT WILL BE GREATER THAN ITMIN.
C***        IF SO, THE TIMES WILL BE SYS1 AND SYS2+THE DIFFERENCE
C***
  100 IT1=GS(FIRST2,NEXT2(FIRST2),NUMGP)-GS(FIRST1,NEXT1(FIRST1),NUMGP)
      IT2=GS(FIRST2,NEXT2(FIRST2),NUMGP)-(GS(FIRST1,NEXT1(FIRST1)+1
     1 ,NUMGP)-1)
      ITIMD=MIN(MAX(IT1,ITMIN),ITMAX)
      IDIFF=0
      IF(ITIMD.EQ.ITMIN) GO TO 130
      IF(ITIMD.LT.IT2) GO TO 120
C***
C***        OUTPUT(1) IS THE DELAY, OUTPUT(2) IS THE SYS2 TIME
C***        OUTPUT 3 AND 4 ARE THE SYS1 AND SYS2 SAT. #'2.
C***        OUTPUT 5 IS THE SYS1 GROUND STATION.
C***        OUTPUT 6 IS THE SYS2 GROUND STATION.
C***
  110 N1O2=NEXT1(FIRST1)/2+1
      N2O2=NEXT2(FIRST2)/2+1
      OUTPUT(NEXT(NUMGP),1)=ITIMD
      OUTPUT(NEXT(NUMGP),2)=(GS(FIRST2,NEXT2(FIRST2),NUMGP)+IDIFF)
      OUTPUT(NEXT(NUMGP),3)=FIRST1
      OUTPUT(NEXT(NUMGP),4)=FIRST2-NSAT1
      OUTPUT(NEXT(NUMGP),5)=IGS(FIRST1,N1O2,NUMGP)
      OUTPUT(NEXT(NUMGP),6)=IGS(FIRST2,N2O2,NUMGP)
      NEXT(NUMGP)=NEXT(NUMGP)+1
C***
C***        OUTPUT IS JUST A BUFFER; IF IT IS FULL, FLUSH IT.
C***
      IF(NEXT(NUMGP).GT.KMAX) CALL OUTBUF
      IFSEE=IFSEE+1
      DIFF=ITIMD*DT/60.
      DIFMAX(IFSEE)=MAX(DIFMAX(IFSEE),DIFF)
      DIFMIN(IFSEE)=MIN(DIFMIN(IFSEE),DIFF)
      DIFFMX=MAX(DIFFMX,DIFF)
      DIFFMN=MIN(DIFFMN,DIFF)
      DIFTOT=DIFTOT+DIFF
      DIFTT(IFSEE)=DIFTT(IFSEE)+DIFF
      NUMDIF=NUMDIF+1
      NUMDF(IFSEE)=NUMDF(IFSEE)+1
      NEXT2(FIRST2)=NEXT2(FIRST2)+2
C***
C***        IF THE SYS1 TIME HAS BEEN REVISITED ENOUGH, GET A NEW
C***        SYS1 TIME AND START AGAIN.
```

```
C***
      IF(IFSEE.LT.MAXSEE) GO TO 70
  120 NEXT1(FIRST1)=NEXT1(FIRST1)+2
      GO TO 40
C***
C***        THIS IS THE SECTION TO CALCULATE DIFFERENCE IF DELAY
C***        IS TMIN
C***
  130 IDIFF=ITMIN-IT1
      IF(IDIFF.LE.(GS(FIRST2,NEXT2(FIRST2)+1,NUMGP)-1)-
     1 GS(FIRST2,NEXT2(FIRST2),NUMGP)) GO TO 110
      NEXT2(FIRST2)=NEXT2(FIRST2)+2
      GO TO 70
      END
```

8.10 Subroutine OUTBUF

```
C***
C******* FLUSHES BUFFER AND RESETS COUNTERS
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      DIMENSION OUT(6)
C***
      DATA LSTART/MAXGP*0/
      DATA DIFLAS/0./
C***
      NFILE=NUMGP+7
      IF(SINGL)NFILE=8
      CONV=DT/60.
```

```
      IF(LSTART(NUMGP).EQ.1) GO TO 20
      TMINN=TMIN/60.
      TMAXX=TMAX/60.
      LSTART(NUMGP)=1
C***
C***        FOR SINGLE FILE OUTPUT WE NEED A LABEL TELLING US WHAT TARGET
C***        THE FOLLOWING OUTPUT IS FOR.
C***
   10 IF(.NOT.SINGL)WRITE(NFILE,70)
      IF(SINGL)WRITE(NFILE,110) ELGS(NUMGP+NGS,1),ELGS(NUMGP+NGS,2)
      IF(SINGL.AND..NOT.SHORT)WRITE(NFILE,120)
   20 CONTINUE
      IF(SHORT)GO TO 50
      I=NUMGP
      NUMFIL=I+7
      IF(SINGL)NUMFIL=8
      NEXT(I)=NEXT(I)-1
      K=NEXT(I)
      DO 40 J=1,K
      OUT(1)=OUTPUT(J,1)*CONV
      OUT(2)=OUTPUT(J,2)*CONV
      OUT(3)=OUTPUT(J,3)
      OUT(4)=OUTPUT(J,4)
      OUT(5)=OUTPUT(J,5)
      OUT(6)=OUTPUT(J,6)
      DIFF=OUT(2)-OUT(1)
      IF(DIFF.EQ.DIFLAS) GO TO 30
      IF((DIFF-DIFLAS).LE.20..AND.ABS(TMAXX-OUT(1)).LE.0.5)
     1 GO TO 30
      WRITE(NUMFIL,90)
      IF(INT(DIFF/1440.)-INT(DIFLAS/1440.).GE.1) WRITE(NUMFIL,100)
   30 WRITE(NUMFIL,80) I,DIFF,OUT(2),OUT(1),OUT(3),
     1 OUT(4),OUT(5),OUT(6)
      NEXT(I)=1
      DIFLAS=DIFF
   40 CONTINUE
   50 CONTINUE
      DO 60 L=1,6
      DO 60 J=1,K
   60 OUTPUT(J,L)=0
      RETURN
   70 FORMAT(1H1,' GP',3X,' SYS1 TIME',2X,' SYS2 TIME',2X,
     1 ' DELAY   SYS1SAT SYS2SAT  SYS1GS  SYS2GS')
   80 FORMAT(1X,I2,3X,3(F8.1,3X),4(F5.0,3X))
   90 FORMAT(1H )
  100 FORMAT(1H0/1H0)
  110 FORMAT(1H0,' FOR TARGET AT LAT=',F6.2,'  LONG=',F6.2)
  120 FORMAT(1H0,' GP',3X,' SYS1 TIME',2X,' SYS2 TIME',2X,
     1 ' DELAY   SYS1SAT SYS2SAT  SYS1GS')
      END
```

AD-A095 537 NAVAL RESEARCH LAB WASHINGTON DC F/G 22/2
LORI: A MULTISATELLITE, MULTISYSTEM, TIME-COVERAGE ANALYSIS PRO--ETC(U)
FEB 81 W H HARR
UNCLASSIFIED NRL-8448 NL

2 OF 2
AD A
095537

END
DATE
FILMED
3-81
DTIC

A

955

8.11 Subroutine SIMUL

```
C***
C***   SIMUL IS THE SIMULTANEOUS COVERAGE PROCESSOR
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,6),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      INTEGER FIRST1,FIRST2,IFSEE,NEXT1,NEXT2,LAST2,G1,G2,G3,GNEXT
      INTEGER ITMIN,ITMAX,ITIMD,IT1,IT2,IDIFF,G1MAX,G2MAX,G3MAX
C***
C***
      DIMENSION NEXT1(MAXSAT),NEXT2(MAXSAT),LAST2(MAXSAT)
      DIMENSION G2(MAXHIT*MAXSAT),G1(MAXHIT*MAXSAT),G3(MAXSAT*MAXHIT)
      DIMENSION SATOT(MAXSAT),GNEXT(2)
C***
      EQUIVALENCE (G1(1),IGS(1,1,1)),(G3(1),GS(1,1,1))
C***
C***
      MAXG1=MAXSAT*MAXHIT
      MAXG2=MAXSAT*MAXHIT
      MAXG3=MAXSAT*MAXHIT
      G1MAX=MAXG1-2
      G2MAX=MAXG2-2
      G3MAX=MAXG3-2
      G1(1)=0
      G1(2)=0
      G1TOT=0.
      G2TOT=0.
      TOTOT=0.
```

```
      CONV=DT/6Ø.
      DO 1Ø I=1,MAXSAT
   1Ø NEXT1(I)=1
      DO 2Ø I=1,MAXSAT
      NEXT2(I)=1
   2Ø CONTINUE
      FIRST1=1
      FIRST2=N1P1
      GNEXT(1)=1
      GNEXT(2)=1
C***
C***   CALCULATE THE TOTAL TIME EACH SAT SAW EACH TARGET
C***
      DO 4Ø J=1,NTOT
      LOOP=KNEXT(J,NUMGP)-2
      TOT=Ø.
      DO 3Ø K=1,LOOP,2
   3Ø TOT=TOT+(GS(J,K+1,NUMGP)-GS(J,K,NUMGP))*CONV
   4Ø SATOT(J)=TOT
      IFILE=NUMGP+7
      IK=NUMGP+NGS
      IF(SINGL)IFILE=8
      ITOT=Ø
      ISUM=Ø
      OPEN(UNIT=IFILE,ACCESS='APPEND',TYPE='OLD')
C***
C***  THIS SECTION PRINTS OUT THE ACTUAL TIMES OF OVERLAP
C***  OF THE TWO SYSTEMS, UNLESS SHORT IS TRUE
C***
      IF(NUMGP.EQ.1.AND.SINGL.AND.DBUG) WRITE(IFILE,56Ø)
      WRITE(IFILE,51Ø) ELGS(IK,1),ELGS(IK,2)
      WRITE(IFILE,53Ø) (SATOT(J),J=1,NTOT)
      I=1
   5Ø CONTINUE
C***
C***    NOW FOR THE SIMULTANEOUS SECTION
C***    THE FOLLOWING SECTION MERGES THE INDIVIDUAL SIGHTINGS OF
C***    THE SATELLITES IN EACH SYSTEM INTO TWO FILES OF SIGHTINGS
C***    OF EACH SYSTEM AS A GROUP.
C***
C***
C***     FIRST FIND FIRST HIT OF SYS1
C***
      DO 6Ø K=1,NSAT1
   6Ø IF(GS(K,NEXT1(K),NUMGP).LT.GS(FIRST1,NEXT1(FIRST1),NUMGP)
     1 )FIRST1=K
C***
C***     IF THE NEXT SYS1 HIT IS PAST THE END, QUIT THIS LOOP
C***
      IF(NEXT1(FIRST1).GE.KNEXT(FIRST1,NUMGP)) GO TO 1ØØ
C***
```

```
C***   IF THE BEGIN OF THE NEXT HIT IS PAST THE END OF THE MERGE FILE
C***   LAST HIT, WE CAN'T MERGE, SO ADD A NEW HIT TO THE MERGE FILE
C***   IF IT ISN'T PAST, MERGE IT INTO THE LAST HIT OF THE MERGE FILE
C***
   70 IF(GS(FIRST1,NEXT1(FIRST1),NUMGP).GT.G1(GNEXT(I)+1)) GO TO 80
      G1(GNEXT(I)+1)=MAX(G1(GNEXT(I)+1),GS(FIRST1,NEXT1(FIRST1)+1,
     1  NUMGP))
      NEXT1(FIRST1)=NEXT1(FIRST1)+2
      GO TO 50
   80 GNEXT(I)=GNEXT(I)+2
      G1(GNEXT(I))=GS(FIRST1,NEXT1(FIRST1),NUMGP)
      G1(GNEXT(I)+1)=GS(FIRST1,NEXT1(FIRST1)+1,NUMGP)
      NEXT1(FIRST1)=NEXT1(FIRST1)+2
      G1(GNEXT(I)+3)=0
      G1(GNEXT(I)+2)=0
C***
C***    MAKE SURE THE MERGE FILE HASN'T GOTTEN TOO BIG
C***
      IF(GNEXT(I).LT.G1MAX) GO TO 90
      TYPE 450
      STOP
   90 CONTINUE
      GO TO 50
  100 CONTINUE
  110 G2(1)=0
      G2(2)=0
C***
C***    NOW DO EXACTLY THE SAME FOR THE SYS2 HITS
C***
  120 I=2
      IF(NSAT2.EQ.0)GO TO 170
C***
C***   FIRST FIND THE FIRST HIT OF THE SYS2 GROUP
C***
      DO 130 K=N1P1,NTOT
  130 IF(GS(K,NEXT2(K),NUMGP).LT.GS(FIRST2,NEXT2(FIRST2),NUMGP))
     1 FIRST2=K
C***
C***    IF WE'RE DONE WITH SYS2, GO TO THE NEXT SECTION
C***
      IF(NEXT2(FIRST2).GE.KNEXT(FIRST2,NUMGP)) GO TO 170
C***
C***    IF NO OVERLAP, ADD A NEW HIT TO MERGE FILE
C***
  140 IF(GS(FIRST2,NEXT2(FIRST2),NUMGP).GT.G2(GNEXT(I)+1)) GO TO 150
C***
C***     OTHERWISE, MERGE IT IN
C***
      G2(GNEXT(I)+1)=MAX(G2(GNEXT(I)+1),GS(FIRST2,NEXT2(FIRST2)+1,
     1 NUMGP))
      NEXT2(FIRST2)=NEXT2(FIRST2)+2
```

```
      GO TO 120
  150 GNEXT(I)=GNEXT(I)+2
      G2(GNEXT(I))=GS(FIRST2,NEXT2(FIRST2),NUMGP)
      G2(GNEXT(I)+1)=GS(FIRST2,NEXT2(FIRST2)+1,NUMGP)
      NEXT2(FIRST2)=NEXT2(FIRST2)+2
      G2(GNEXT(I)+2)=0
      G2(GNEXT(I)+3)=0
      IF(GNEXT(I).LT.G2MAX) GO TO 160
      TYPE 460
      STOP
  160 CONTINUE
      GO TO 120
  170 CONTINUE
C***
C*** TOTAL UP THE TOTAL COVERAGE TIME OF EACH SYSTEM
C***
      DO 180 J=1,GNEXT(1),2
  180 G1TOT=G1TOT+(G1(J+1)-G1(J))*CONV
      DO 190 J=1,GNEXT(2),2
  190 G2TOT=G2TOT+(G2(J+1)-G2(J))*CONV
C***
C*** NOW HAVE TWO ARRAYS OF START-STOP TIMES,
C*** AND ONE FOR SYS2
C*** NOW HAVE TO COMPARE THESE TWO TO GET THE SIMULTANEOUS TIMES.
C*** THE SYS1 AND SYS2 FILES ARE EITHER MERGED OR SORTED,
C*** DEPENDING ON METHOD.
C*** UNLESS THE SYSTEM HAD A START TIME OF ZERO(HIT ON FIRST STEP),
C*** THE FIRST PAIR OF ELEMENTS
C*** WILL BE ZERO.  THE FIRST BUSINESS THAT FOLLOWS IS TO TAKE THAT
C*** INTO ACCOUNT BY NOT STARTING WITH THE FIRST ELEMENTS UNLESS
C*** NECESSARY
C***
      N1=1
      N2=1
      N3=1
      IF(G1(2).EQ.0)N1=3
      IF(G2(2).EQ.0)N2=3
C***
C***    FIRST OF SYS2 HAS TO BE LESS THAN LAST OF SYS1
C***
  200 IF(G2(N2).LT.G1(N1+1)) GO TO 220
C***
C***    IF IT WASN'T, INCREMENT SYS1, DECREMENT SYS2
C***
  210 N1=N1+2
      N2=N2-2
      N2=MAX(1,N2)
      IF(G2(N2+1).EQ.0) N2=3
      IF(N1.GE.GNEXT(1)+2) GO TO 250
      GO TO 200
C***
```

```
C***    FIRST OF SYS1 HAS TO BE LESS THAN LAST OF SYS2
C***
  220 IF(G1(N1).LT.G2(N2+1)) GO TO 230
C***
C***   IF IT WASN'T, INCREMENT SYS2
C***
      N2=N2+2
      IF(N2.GE.GNEXT(2)+2) GO TO 210
      GO TO 200
C***
C***WE HAVE AN OVERLAP-NOW FIGURE THE TIME OF OVERLAP AND
C*** WRITE IT OUT
C***
  230 IST=MAX(G1(N1),G2(N2))
      ISTP=MIN(G1(N1+1),G2(N2+1))
      G3(N3)=IST
      G3(N3+1)=ISTP
C***
C***   INCREMENT OUTPUT COUNTER AND SYS2 COUNTER
C***
      N3=N3+2
      N2=N2+2
      G3(N3)=32000
      G3(N3+1)=32000
      IF(N3.LT.G3MAX) GO TO 240
      TYPE 470,N3,G3MAX
      STOP
  240 CONTINUE
      GO TO 200
  250 CONTINUE
      IF(SHORT) GO TO 270
      L=N3-2
      IF(L.LE.0) GO TO 290
      WRITE(IFILE,480)
      DO 260 J=1,L,2
      TONE=G3(J)*CONV
      TTWO=G3(J+1)*CONV
      TDIFF=TTWO-TONE
      WRITE(IFILE,500) TONE,TTWO,TDIFF
  260 CONTINUE
  270 L=N3-2
      IF(L.LE.0) GO TO 290
      DO 280 J=1,L,2
      ISUM=G3(J+1)-G3(J)
      ITOT=ITOT+ISUM
  280 CONTINUE
  290 CONTINUE
C***
C***   TOT IS TOTAL SIMULTANEOUS COVERAGE DURATION
C***   G1TOT,G2TOT ARE EACH SYSTEMS TOTAL COVERAGE TIME
C***   SATOT() IS THE INDIVIDUAL SATELLITE TOTAL COVERAGE TIME
```

```
C***
      TOT=ITOT*CONV
      WRITE(IFILE,490) TOT
      WRITE(IFILE,520) G1TOT,G2TOT
C***
C***  THIS SECTION WRITES OUT EACH SYSTEMS ACTUAL COVERAGE TIMES
C***
      IF(.NOT.SYSTM) GO TO 310
      WRITE(IFILE,550)
      LOOPND=MAX(GNEXT(1),GNEXT(2))
      INC1=0
      IF(G1(2).EQ.0)INC1=2
      INC2=0
      IF(G2(2).EQ.0)INC2=2
      DO 300 ICOUNT=1,LOOPND,2
      S1A=G1(ICOUNT+INC1)*CONV
      S1B=G1(ICOUNT+INC1+1)*CONV
      S2A=G2(ICOUNT+INC2)*CONV
      S2B=G2(ICOUNT+INC2+1)*CONV
      IF(ICOUNT+INC1.GT.GNEXT(1))S1A=0.
      IF(ICOUNT+INC1.GT.GNEXT(1))S1B=0.
      IF(ICOUNT+INC2.GT.GNEXT(2))S2A=0.
      IF(ICOUNT+INC2.GT.GNEXT(2))S2B=0.
      WRITE(IFILE,540) S1A,S1B,S2A,S2B
  300 CONTINUE
  310 CONTINUE
      IF(.NOT.TOTAL) GO TO 440
      WRITE(IFILE,570)
      IF(.NOT.SHORT)WRITE(IFILE,580)
      INUM=0
      MAXDUR=-1
      MINDUR=LARGE
      MAXDEL=-1
      MINDEL=LARGE
      AVDEL=0.
      AVDUR=0.
      IDELT=0
      IDURT=0
      IOUT1=0
      IOUT2=0
      NXT1=1
      NXT2=1
      IFLG1=0
      IFLG2=0
      IFLG=0
      IF(G1(2).EQ.0)NXT1=3
      IF(G2(2).EQ.0)NXT2=3
  320 IFLG=IFLG1+2*IFLG2
      GO TO (360,340,430)IFLG
  330 IF(G1(NXT1).GT.G2(NXT2))GO TO 360
  340 LOC=1
```

```
      IF(G1(NXT1).GT.IOUT2) GO TO 380
      IOUT2=MAX(G1(NXT1+1),IOUT2)
  350 NXT1=NXT1+2
      IF(NXT1.GT.GNEXT(1))IFLG1=1
      GO TO 320
  360 LOC=2
      IF(G2(NXT2).GT.IOUT2)GO TO 380
      IOUT2=MAX(G2(NXT2+1),IOUT2)
  370 NXT2=NXT2+2
      IF(NXT2.GT.GNEXT(2))IFLG2=1
      GO TO 320
  380 IF(IOUT2.EQ.0) GO TO 400
      INUM=INUM+1
      IDUR=IOUT2-IOUT1
      IDURT=IDURT+IDUR
      MAXDUR=MAX(MAXDUR,IDUR)
      MINDUR=MIN(MINDUR,IDUR)
      IDEL=IOUT1-IOLD2
      IF(IDEL.EQ.0)GO TO 390
      IDELT=IDELT+IDEL
      MAXDEL=MAX(MAXDEL,IDEL)
      MINDEL=MIN(MINDEL,IDEL)
  390 IF(.NOT.SHORT)WRITE(IFILE,620)CONV*IOUT1,CONV*IOUT2,
     1 CONV*IDUR,CONV*IDEL
  400 IOLD1=IOUT1
      IOLD2=IOUT2
      GO TO (410,420)LOC
  410 IOUT1=G1(NXT1)
      IOUT2=G1(NXT1+1)
      GO TO 350
  420 IOUT1=G2(NXT2)
      IOUT2=G2(NXT2+1)
      GO TO 370
  430 IDEL=IOUT1-IOLD2
      IDUR=IOUT2-IOUT1
      IF(.NOT.SHORT)WRITE(IFILE,620)CONV*IOUT1,CONV*IOUT2,
     1 CONV*IDUR,CONV*IDEL
      IDELT=IDELT+IDEL
      IDURT=IDURT+IDUR
      MAXDUR=MAX(MAXDUR,IDUR)
      MINDUR=MIN(MINDUR,IDUR)
      MAXDEL=MAX(MAXDEL,IDEL)
      MINDEL=MIN(MINDEL,IDEL)
      INUM=INUM+1
      AVDEL=(CONV*IDELT)/INUM
      AVDUR=(CONV*IDURT)/INUM
      WRITE(IFILE,590)MAXDUR*CONV,MINDUR*CONV,AVDUR,INUM
      WRITE(IFILE,600)CONV*MAXDEL,CONV*MINDEL,AVDEL
      WRITE(IFILE,610)CONV*IDURT
  440 CONTINUE
      CLOSE(UNIT=IFILE)
```

```
      RETURN
  450 FORMAT(' TOO MANY HITS IN SIMUL FOR SYS1')
  460 FORMAT(' TOO MANY HITS IN SIMUL FOR SYS2')
  470 FORMAT(' OVERFLOW IN SIMUL,N3=',I5,' G3MAX=',I5)
  480 FORMAT(1H ,' START OF  COVERAGE  END OF MUTUAL COVERAGE',
     1 '    PERIOD OF COVERAGE(IN MIN.)')
  490 FORMAT(' TOTAL TIME OF SIMULTANEOUS COVERAGE IS ',F10.2,' MIN.')
  500 FORMAT(5X,F10.1,10X,F10.1,10X,F10.1)
  510 FORMAT('0 SIMULTANEOUS COVERAGE FOR TARGET ',F9.2,2X,F9.2)
  520 FORMAT(' TOTAL COVERAGE FOR SYS1 AND SYS2 IS ',
     1 F9.2,3X,F9.2)
  530 FORMAT(' TOTAL COVERAGE FOR EACH SATELLITE IS ',/,
     1 10(F9.2,3X) )
  540 FORMAT(4X,4(F10.2,3X))
  550 FORMAT('     SYS1STRT        SYS1STP        SYS2STRT        SYS2STP  ')
  560 FORMAT(1H1)
  570 FORMAT('  TOTAL MERGED COVERAGE FOR BOTH SYSTEMS ')
C 570 FORMAT('0          START          STOP')
  580 FORMAT(1H0,9X,'START',10X,'STOP',6X,'DURATION',9X,'DELAY')
  590 FORMAT('  MAXIMUM TOTAL SIGHTING DURATION = ',F10.2,/
     1 ,'  MINIMUM TOTAL SIGHTING DURATION = ',F10.2,/,
     2  '  AVERAGE TOTAL SIGHTING DURATION = ',F10.2,/,
     3  '  TOTAL NUMBER OF MERGED SIGHTINGS = ',I8)
  600 FORMAT('  MAXIMUM TOTAL SIGHTING DELAY = ',F10.2,/,
     1  '  MINIMUM TOTAL SIGHTING DELAY = ',F10.2,/,
     2  '  AVERAGE TOTAL SIGHTING DELAY = ',F10.2)
  610 FORMAT(' TOTAL TIME TARGET SEEN BY BOTH SYSTEMS = ',F12.2,/)
  620 FORMAT(3X,F12.2,2X,F12.2,2X,F12.2,2X,F12.2)
      END
```

8.12 Subroutine MATRIX

```
C***
C***   MATRIX CALCULATES AND PRINTS OUT THE HIT MATRIX FOR ALL THE
C***   SATELLITES AND TARGETS.  THIS MATRIX CONSISTS OF 1'S FOR HITS
C***   AND BLANKS FOR NON-HITS, REFERENCED BY SATELLITE AND TIME PERIOD.
C***
      INTEGER GS,OUTPUT,DELTAP,NUMP,PSTART,AFRAC
      LOGICAL SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM
      LOGICAL NUMHIT,VARIA,IFDONE,ALLDUN,MAT,TOTAL,MEAN,SFIRST
C***
      PARAMETER MAXGP=  4,MAXG=  8,MAXSAT=  4,MAXHIT= 800
C***
      COMMON/TRUTH/SIMULT,SINGL,SHORT,GRID,DBUG,DELA,SYSTM,
     1 NUMHIT,VARIA,IFDONE(MAXGP),ALLDUN,MAT,TOTAL,MEAN,SFIRST(MAXSAT)
      COMMON/ELG/ELG(MAXG,6),ELGS(MAXG,4),HITAB(MAXSAT,MAXGP)
     1 ,TRG(MAXGP,5)
      COMMON/GS/KNEXT(MAXSAT,MAXGP),MAXGS,GS(MAXSAT,MAXHIT,MAXGP),
     1 IGS(MAXSAT,MAXHIT/2,MAXGP)
      COMMON/ELSAT/ELSAT(MAXSAT,16)
      COMMON/GRIDIN/BLAT,BLON,ELAT,ELON,SPALAT,SPALON,NUMLAT,NUMLON,
     1 NUMTAR,LASTK,LASTI,LASTJ,RADII
      COMMON/NUM/NSAT1,NSAT2,NGP,NGS,N1P1,NTOT,NGSP1,NGTOT
      COMMON/TIME/TIME,TSTOP,DT,TMIN,TMAX,ISTOP(MAXGP)
      COMMON/OUTPUT/OUTPUT(100,5),NEXT(MAXGP)
      COMMON/CONS/DTR,RTD,PI,TPI,RE,XMU,XJ2,WE,LARGE
      COMMON/FLAG/IFLAG(MAXSAT,MAXGP),KMAX,INC,IFHTGP(MAXGP),
     1  NUMGP,LSTART(MAXGP),INP,NUMGS(MAXSAT,MAXGP)
      COMMON/DELTOT/DIFFMX,DIFFMN,DIFTOT,NUMDIF,DIFMAX(20),DIFMIN(20),
     1  DIFTT(20),NUMDF(20)
      COMMON/MATRX/PSTART,DELTAP,NUMP,AFRAC
C***
      INTEGER PS,PE,HS,HE,HITS(MAXSAT),HITSAT(MAXSAT)
      INTEGER HITPRT(2),HCOUNT(MAXSAT),HITSUM
C***
C***
      HITPRT(2)='1'
      HITPRT(1)=' '
      DO 100 ITARG=1,NGP
      IFILE=7+ITARG
      IF(SINGL)IFILE=8
      OPEN(UNIT=IFILE,TYPE='OLD',ACCESS='APPEND')
      WRITE(IFILE,110)ITARG,ELGS(ITARG,1),ELGS(ITARG,2)
      WRITE(IFILE,120)
      WRITE(IFILE,130)(I,I=1,NTOT)
      PQUIT=NUMP*DELTAP+PSTART
      DO 10 I=1,NTOT
      HITSAT(I)=0
   10 HCOUNT(I)=1
      PS=PSTART-DELTAP
      PE=PSTART
   20 PS=PE
```

```
      PE=PS+DELTAP
      IF(PS.GE.PQUIT) GO TO 90
      DO 70 I=1,NTOT
      HITS(I)=1
      HITSUM=0
   30 IF(HCOUNT(I).GT.KNEXT(I,ITARG))GO TO 70
      HS=GS(I,HCOUNT(I),ITARG)
      HE=GS(I,HCOUNT(I)+1,ITARG)
      IF(PE.LE.HS)GO TO 70
      IF(PS.LT.HE)GO TO 40
      HCOUNT(I)=HCOUNT(I)+2
      GO TO 30
   40 IF(PE.GT.HE) GO TO 50
      IF(PE-HS.GE.AFRAC)HITS(I)=2
      GO TO 70
   50 IF(PS.GE.HS)GO TO 60
      IF(HE-HS.GE.AFRAC)HITS(I)=2
      GO TO 70
   60 IF((HE-PS).GE.AFRAC)HITS(I)=2
      GO TO 70
   70 CONTINUE
      DO 80 I=1,NTOT
      HITSAT(I)=HITSAT(I)+HITS(I)-1
   80 HITSUM=HITSUM+HITS(I)-1
      IPER=(PS-PSTART)/DELTAP+1
      WRITE(IFILE,140)IPER,(HITPRT(HITS(K)),K=1,NTOT),HITSUM
      GO TO 20
   90 WRITE(IFILE,150)(HITSAT(I),I=1,NTOT)
      WRITE(IFILE,160)
      CLOSE(UNIT=IFILE)
  100 CONTINUE
      RETURN
  110 FORMAT('1HIT MATRIX FOR TARGET # ',I4,' AT LAT, LONG = '
     1  ,'(',F7.2,',',F7.2,')')
  120 FORMAT(' PER #     SAT. # ')
C***
C***   NOTE: THE FOLLOWING THREE FORMATS USE THE NON-STANDARD
C***   <#> VARIABLE FORMAT STATEMENT.  THIS CAN BE CIRCUMVENTED
C***   IN NON-DEC MACHINES BY USING ARRAYS FOR FORMAT STATEMENTS
C***   OR BY JUST REPLACING THE <NTOT> WITH SOME REASONABLE NUMBER.
C***   IT IS LEFT IN THIS MANNER HERE AS IT LEAVES THE OUTPUT IN
C***   A MUCH MORE READABLE MANNER.
C***
  130 FORMAT(5X,<NTOT>I3,3X,'TOT.# SATS')
  140 FORMAT(1H ,I3,1X,<NTOT>(2X,A1),5X,I2)
  150 FORMAT(1H0,4X,<NTOT>I3)
  160 FORMAT(1H0,'TOTAL NUMBER OF PERIODS PER SAT')
      END
```

8.13 Subroutine MULTI

```
C***
C***   MULTI IS A SAMPLE POST-PROCESSOR,
C***   DESIGNED TO WRITE OUT EACH TIME THE CONSTELLATION OF SATELLITES
C***   IN VIEW OF THE TARGET CHANGES.  IT WRITES OUT THE START AND STOP
C***   TIMES THE SATELLITES WERE IN VIEW, AND LISTS THE SATELLITES IN
C***   VIEW.  IT HAS AS INPUT MINSATS, THE MINIMUM NUMBER OF SATELLITES
C***   IN VIEW.  MULTI WILL WRITE OUT ONLY THOSE TIMES WHEN THE NUMBER
C***   OF SATELLITES IN VIEW IS MINSATS OR GREATER.
C***
      INCLUDE 'INCLUDE.INC'
C***
      INTEGER FINTOT,SNEXT(MAXSAT)
      LOGICAL FIN(MAXSAT)
      CHARACTER*(MAXSAT) PICT,PICTO,BLANK
      INTEGER IZERO
C***
C***      COMMON/MULTIP/MINSATS
C***
C***
      DATA FIN/MAXSAT*.FALSE./
      DATA IZERO/Ø/
      DATA BLANK/' '/
C***
C***    INITIALIZATION
C***
      ITIM=-1
      ITIMO=Ø
      CONV=DT/6Ø.
      FINTOT=Ø
      NUMFIL=7+NUMGP
      IF(SINGL)NUMFIL=8
      OPEN(UNIT=NUMFIL,TYPE='OLD',ACCESS='APPEND')
      WRITE(NUMFIL,11Ø)
      WRITE(NUMFIL,12Ø) NUMGP
      WRITE(NUMFIL,13Ø) ELGS(NUMGP,1),ELGS(NUMGP,2)
      WRITE(NUMFIL,14Ø) MINSATS
      WRITE(NUMFIL,15Ø)(I,I=1,NTOT)
      DO 1Ø I=1,NTOT
      PICT(I:I)=' '
      PICTO(I:I)=' '
      FIN(I)=.FALSE.
   1Ø SNEXT(I)=1
C***
C***     LOOP OVER ALL TIME STEPS
C***
   2Ø ITIM=ITIM+1
      DO 3Ø I=1,NTOT
   3Ø PICT(I:I)=' '
      IF(ITIM.GT.TSTOP/DT) GO TO 9Ø
C***
```

```
C***   AT EACH TIME STEP, LOOP OVER ALL SATELLITES
C***   IF THE TIME STEP IS WITHIN A START-STOP PAIR FOR A SATELLITE,
C***   ADD ITT TO THE PICTURE.
C***
      DO 7Ø ISAT=1,NTOT
   4Ø CONTINUE
      IF(FIN(ISAT)) GO TO 7Ø
      IF (ITIM.GE.GS(ISAT,(SNEXT(ISAT)+1),NUMGP)) GO TO 5Ø
      IF(ITIM.GE.GS(ISAT,SNEXT(ISAT),NUMGP).AND.ITIM.LT.
     1     GS(ISAT,SNEXT(ISAT)+1,NUMGP))PICT(ISAT:ISAT)='*'
      GO TO 7Ø
   5Ø SNEXT(ISAT)=SNEXT(ISAT)+2
      IF(SNEXT(ISAT).LT.KNEXT(ISAT,NUMGP))GO TO 6Ø
      FIN(ISAT)=.TRUE.
      FINTOT=FINTOT+1
      IF(FINTOT.GE.NTOT) GO TO 9Ø
   6Ø GO TO 4Ø
   7Ø CONTINUE
      FINTOT=Ø
C***
C***   IF THE SATELLITES IN VIEW HAS CHANGED FROM THE LAST TIME STEP,
C***   WRITE OUT THE OLD PICTURE(AS LONG AS THE # OF SATS IS GREATER
C***   THAN MINSATS)
C***
      IF(PICT.EQ.PICTO)GO TO 2Ø
      NUMSATS=NUMOCC('*',PICTO)
      IF(NUMSATS.GE.MINSATS)
     1 WRITE(NUMFIL,16Ø)ITIMO*CONV,(ITIM-1)*CONV,(PICTO(J:J),J=1,NTOT)
     2  ,NUMSATS
      PICTO=PICT
      ITIMO=ITIM
      DO 8Ø J=1,NTOT
   8Ø PICT(J:J)=' '
      GO TO 2Ø
   9Ø CONTINUE
C***
C***   WE ARE AT THE END.  IF SATS WERE IN VIEW AT END, WRITE OUT
C***   THE CURRENT PICTURE.
C***
      IF(ITIM.GE.TSTOP/DT)  GO TO 1ØØ
      NUMSATS=NUMOCC('*',PICT)
      IF(NUMSATS.GE.MINSATS)WRITE (NUMFIL,16Ø)
     1  ITIMO*CONV,TSTOP/DT*CONV,(BLANK(J:J),J=1,NTOT),IZERO
  1ØØ CLOSE(UNIT=NUMFIL)
      RETURN
  11Ø FORMAT('1 MULTI-SATELLITE COVERAGE LIST')
  12Ø FORMAT('  FOR TARGET NUMBER ',I4)
  13Ø FORMAT(' AT LAT, LONG = (',F1Ø.2,',',F1Ø.2,')')
  14Ø FORMAT(' ALL TIMES WITH ',I3,' OR MORE SATELLITES IN VIEW'
     1 ,/,' AT THE SAME TIME ARE LISTED')
  15Ø FORMAT('ØSTART-TIME   STOP-TIME',<NTOT>(2X,I2),'  TOT. #')
```

```
  160 FORMAT(1X,F10.2,2X,F10.2,<NTOT>(3X,A1),3X,I3)
      END
      FUNCTION NUMOCC(CHAR,STRING)
C***
C***    FUNCTION TO TELL THE NUMBER OF OCCURANCES OF A STRING IN
C***    ANOTHER STRING
C***    CHAR IS THE STRING BEING LOOKED FOR,
C***    STRING IS THE STRING BEING SEARCHED
C***
      CHARACTER*(*) CHAR,STRING
      L=LEN(CHAR)
      NUM=0
      DO 10 I=1,LEN(STRING)
      IF(CHAR.EQ.STRING(I:I+L-1))NUM=NUM+1
   10 CONTINUE
      NUMOCC=NUM
      RETURN
      END
```

MULTI-SATELLITE COVERAGE LIST
FOR TARGET NUMBER 1
AT LAT, LONG = (30.00, 270.00)
ALL TIMES WITH 1 OR MORE SATELLITES IN VIEW
AT THE SAME TIME ARE LISTED

START-TIME	STOP-TIME	1	2	3	4	TOT. #
112.00	116.00				*	1
436.00	440.00			*		1
497.00	503.00		*			1
549.00	561.00	*				1
605.00	617.00		*			1
662.00	672.00	*				1
723.00	726.00		*			1
938.00	942.00				*	1
1066.00	1073.00		*			1
1120.00	1132.00	*				1
1175.00	1188.00		*			1
1232.00	1242.00	*				1
1508.00	1512.00				*	1
1947.00	1957.00	*				1
2001.00	2014.00		*			1
2057.00	2069.00	*				1
2116.00	2124.00		*			1
2465.00	2466.00		*			1
2517.00	2527.00	*				1
2571.00	2584.00		*			1
2627.00	2640.00	*				1
2686.00	2692.00		*			1
2770.00	2773.00			*		1
3340.00	3344.00			*		1
3346.00	3351.00	*				1
3398.00	3410.00		*			1
3453.00	3466.00	*				1
3511.00	3521.00		*			1
3571.00	3575.00	*				1
3842.00	3845.00				*	1
3914.00	3921.00	*				1
3968.00	3980.00		*			1
4023.00	4036.00	*				1
4080.00	4091.00		*			1
4166.00	4170.00			*		1
4412.00	4416.00				*	1
4737.00	4741.00			*		1
4796.00	4805.00		*			1
4849.00	4862.00	*				1
4906.00	4918.00		*			1
4964.00	4972.00	*				1
5238.00	5243.00				*	1
5365.00	5375.00		*			1
5420.00	5433.00	*				1
5476.00	5488.00		*			1
5534.00	5541.00	*				1
5563.00	5567.00			*		1

8.14 Program REDIM

```
C
C  PROGRAM TO CALCULATE DIMENSIONS OF THE ARRAYS IN PROGRAM LORI
C  AND TO WRITE THE COMMAND FILE TO EDIT THE SUBROUTINES OF LORI
C
      DOUBLE PRECISION FIL1(2),FIL2(2),FIL3(2),FIL4(2),FIL5(2),FIL6(2)
      DOUBLE PRECISION FIL7(2)
      DATA FIL1/'LORI.FOR','        '/
      DATA FIL2/'INPUT.FO','R       '/
      DATA FIL3/'SETUP.FO','R       '/
      DATA FIL4/'COREL8.F','OR      '/
      DATA FIL5/'DELAY.FO','R       '/
      DATA FIL6/'SIMUL.FO','R       '/
      DATA FIL7/'MATRIX.F','OR      '/
C     DATA FIL8/'POSTPROCESSOR NAME HERE'/
      OPEN(UNIT=1,CARRIAGECONTROL='LIST',NAME='B.COM',TYPE='NEW')
   10 CONTINUE
      TYPE 20
   20 FORMAT(' INPUT MAXIMUM # OF  SATS, MAXIMUM # OF TARGETS,'
     1 ,/,' AND MAXIMUM # OF GROUND STATIONS.'
     2 ,/,'   I WILL MAXIMIZE FOR LENGTH OF RUN .'
     3 ,/, '   IF THE VALUES ARE NOT WHAT YOU WANT YOU MAY ',
     4 'INPUT YOUR OWN')
      ACCEPT *,NSAT,NTARG,NGSTA
      NG=NTARG+NGSTA
      NA=23*NSAT+7*NG+59*NTARG+NSAT*NG
      NA=17700-NA
      NA=NA/(NSAT*NTARG)
      NA=(4*(NA-2))/5
      NHIT=NA
C***
C***   THE ABOVE ALGORITHM WAS DESIGNED TO LET THE PROGRAM FIT ON
C***   A DEC PDP 11/70.  IT CAN BE MODIFIED TO USE ON OTHER MACHINES.
C***   IT IS BASED MOSTLY ON TRIAL-AND-ERROR.
C***
      TYPE 30,NSAT,NTARG,NGSTA,NHIT
   30 FORMAT(' OKAY, WE ARE SET TO RUN WITH THE FOLLOWING DIMENSIONS'
     1 ,/,' # SATS = ',I3,' # TARGETS = ',I3,' # G. STA. = ',I3,
     2 /,'   # HITS = ',I4)
      TYPE 40
   40 FORMAT(' IF YOU WANT TO CONTINUE TYPE A 0,ELSE A 1')
      ACCEPT *,IFCONT
      IF(IFCONT.NE.0) GO TO 10
      TYPE 50
   50 FORMAT(' TYPE IN # OF SATS, # OF TARGETS, # OF STATIONS'
     1 /,'  AND # OF HITS ')
      ACCEPT *,MAXSAT,MAXGP,NGSTA,MAXHIT
      MAXG=NGSTA+MAXGP
      MAXGS=MAXSAT*MAXHIT*MAXGP
      MAXIGS=MAXSAT*(MAXHIT/2)*MAXGP
      WRITE(1,60) FIL1
```

```
   60 FORMAT('$EDIT/SLP ',2A8,'/NOAUDIT_TRAIL ')
      WRITE(1,70)
   70 FORMAT('-/PARAMETER MAX/,.')
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
   80 FORMAT('      PARAMETER MAXGP=',I3,',MAXG=',I3,',MAXSAT=',
     1 I3,',MAXHIT=',I4)
      WRITE(1,90)
   90 FORMAT('/')
      WRITE(1,60) FIL2
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,90)
      WRITE(1,60) FIL3
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,90)
      WRITE(1,60) FIL4
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,90)
      WRITE(1,60) FIL5
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,90)
      WRITE(1,60)FIL6
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
C       WRITE(1,70)
C       WRITE(1,100) MAXG1,MAXG2,MAXG3
      WRITE(1,90)
      WRITE(1,60)FIL7
      WRITE(1,70)
      WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
      WRITE(1,90)
C***
C***  IF A POST-PROCESSOR IS ADDED TO LORI, THE FOLLOWING SHOULD
C***  BE UN-COMMENTED TO ALLOW REDIM TO EDIT IT TO PUT IN THE
C***  CORRECT DIMENSIONING INFORMATION.
C***
C WRITE(1,60)FIL8
C WRITE(1,70)
C WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
C WRITE(1,70)
```

```
C WRITE(1,80)MAXGP,MAXG,MAXSAT,MAXHIT
  100 FORMAT('      PARAMETER MAXG1=',I6,',MAXG2=',I6,',MAXG3=',I6)
C      WRITE(1,110)
      WRITE(1,110) FIL1
      WRITE(1,110) FIL2
      WRITE(1,110) FIL3
      WRITE(1,110) FIL4
      WRITE(1,110) FIL5
      WRITE(1,110) FIL6
      WRITE(1,110) FIL7
C WRITE(1,130) FIL8
  110 FORMAT('$PURGE ',2A8)
      WRITE(1,120)
  120 FORMAT('$@C')
      CLOSE (UNIT=1)
      OPEN (UNIT=2,TYPE='NEW', NAME='C.COM',CARRIAGECONTROL='LIST')
      WRITE(2,130)FIL1
      WRITE(2,130)FIL2
      WRITE(2,130)FIL3
      WRITE(2,130)FIL4
      WRITE(2,130)FIL5
      WRITE(2,130)FIL6
      WRITE(2,130)FIL7
C WRITE(2,150)FIL8
  130 FORMAT('$FORTRAN ',2A8)
      WRITE(2,140)
  140 FORMAT('$LINK/MAP:LORI LORI,INPUT,SETUP,COREL8,DELAY,SIMUL-'
     1  ,/,' ,MATRIX,[UTILITY]VAXTIME')
C***
C***   FOR THE PDP 11/70 THE FOLLOWING IS REQUIRED.
C***   THE LINK STATEMENT WOULD BE CHANGED TO REFER TO AN OVERLAY FILE
C***
C      WRITE(2,1200)
  150 FORMAT('ACTFIL=2')
      NUMACT=MAXGP+7
C      WRITE(2,1300) NUMACT
  160 FORMAT('UNITS=',I3)
C      WRITE(2,600)
      WRITE(2,170)
      WRITE(2,180)
      WRITE(2,190)
      WRITE(2,200)
      WRITE(2,210)
      WRITE(2,220)
      WRITE(2,230)
  170 FORMAT('$DELETE LORI.OBJ;*')
  180 FORMAT('$DELETE INPUT.OBJ;*')
  190 FORMAT('$DELETE SETUP.OBJ;*')
  200 FORMAT('$DELETE COREL8.OBJ;*')
  210 FORMAT('$DELETE DELAY.OBJ;*')
  220 FORMAT('$DELETE SIMUL.OBJ;*')
```

```
230 FORMAT('$DELETE MATRIX.OBJ;*')
    WRITE(2,240)
240 FORMAT('$PURGE LORI.EXE,LORI.MAP')
    WRITE(2,250)
250 FORMAT('$PURGE B.COM,C.COM')
    CLOSE(UNIT=2)
    END
```

9.0 ACKNOWLEDGMENT

The author would like to thank Mr. Bernard Kaufman for the many discussions concerning the form and content of the program, and for his help with this writeup.

ATE

LME